

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

SAMSUNG ELECTRONICS CO., LTD.;
SAMSUNG ELECTRONICS AMERICA, INC.,
Petitioner

v.

NETWORK-1 TECHNOLOGIES, INC.,
Patent Owner.

Case No. IPR2025-0118
U.S. Patent No. 12,207,094

**PETITION FOR *INTER PARTES* REVIEW
OF U.S. PATENT NO. 12,207,094**

TABLE OF CONTENTS

	Page
I. INTRODUCTION	1
II. MANDATORY NOTICES UNDER 37 C.F.R. §42.8.....	1
III. FEE AUTHORIZATION	3
IV. GROUNDS FOR STANDING.....	3
V. PRECISE RELIEF REQUESTED	3
VI. THE CHALLENGED PATENT	4
VII. PROSECUTION HISTORY OF THE '094 PATENT.....	8
VIII. LEVEL OF ORDINARY SKILL IN THE ART.....	9
IX. PRIORITY DATE	9
X. CLAIM CONSTRUCTION	10
XI. BRIEF DESCRIPTION OF THE APPLIED PRIOR ART	11
A. Park (Ex-1005)	11
B. GlobalPlatform (Ex-1006).....	12
C. AbiChar (Ex-1007).....	16
D. X9.63-Overview (Ex-1008)	16
E. Weiss (Ex-1009).....	18
F. Nix175 (Ex-1016).....	18
XII. DETAILED EXPLANATION OF THE UNPATENTABILITY GROUNDS.....	18
A. Ground 1: Claims 1-17 and 19-22 are obvious over Park in view of GlobalPlatform and AbiChar.	18
1. A POSITA would have been motivated to combine Park's teachings with GlobalPlatform's and AbiChar's teachings and would have had a reasonable expectation of success.....	19
a. A POSITA would have turned to GlobalPlatform for implementation-level details of Park's security requirements.	19

TABLE OF CONTENTS

(continued)

	Page
<ul style="list-style-type: none"> <ul style="list-style-type: none"> <ul style="list-style-type: none"> b. A POSITA would have understood that Abi-Char's elliptic curve Diffie-Hellman key agreement could be implemented within Park's SPA to provide an efficient asymmetric key-exchange mechanism..... 	25
2. Independent Claim 1	27
<ul style="list-style-type: none"> <ul style="list-style-type: none"> <ul style="list-style-type: none"> a. Element 1[pre]: A method comprising: 	27
<ul style="list-style-type: none"> <ul style="list-style-type: none"> <ul style="list-style-type: none"> b. Element 1[a][1]: (a) generating a first message comprising 	28
<ul style="list-style-type: none"> <ul style="list-style-type: none"> <ul style="list-style-type: none"> c. Element 1[a][2][i]: (1) an identity of an embedded universal integrated circuit card (eUICC); 	29
<ul style="list-style-type: none"> <ul style="list-style-type: none"> <ul style="list-style-type: none"> d. Element 1[a][2][ii]: (2) a nonce; and..... 	30
<ul style="list-style-type: none"> <ul style="list-style-type: none"> <ul style="list-style-type: none"> e. Element 1[a][2][iii]: (3) a first digital signature, generated using a first eUICC private key, wherein the first eUICC private key corresponds to a first eUICC public key; 	32
<ul style="list-style-type: none"> <ul style="list-style-type: none"> <ul style="list-style-type: none"> f. Element 1[b]: (b) sending the first message via a network application to a subscription manager system; 	32
<ul style="list-style-type: none"> <ul style="list-style-type: none"> <ul style="list-style-type: none"> g. Element 1[c]: (c) deriving a second eUICC private key and a corresponding second eUICC public key using a first random number generator and a first set of cryptographic algorithms; 	34
<ul style="list-style-type: none"> <ul style="list-style-type: none"> <ul style="list-style-type: none"> h. Element 1[d]: (d) storing a subscription manager public key which corresponds to a subscription manager private key; 	35
<ul style="list-style-type: none"> <ul style="list-style-type: none"> <ul style="list-style-type: none"> i. Element 1[e][1]: (e) deriving a profile key using a key exchange algorithm based on at least: 	37
<ul style="list-style-type: none"> <ul style="list-style-type: none"> <ul style="list-style-type: none"> j. Element 1[e][2]: wherein the profile key can also be derived at the subscription manager system based at least on: the second eUICC public key, and the subscription manager private key; 	38

TABLE OF CONTENTS

(continued)

	Page
k. Element 1[f]: receiving from the subscription manager system, an encrypted profile comprising a ciphertext including a key K encrypted with a symmetric key;.....	38
l. Element 1[g]: (g) receiving the symmetric key;.....	41
m. Element 1[h]: (h) decrypting, by the embedded universal integrated circuit card, at least a portion of the encrypted profile using the profile key;	42
n. Element 1[i]: (i) decrypting, by the embedded universal integrated circuit card, at least a portion of the ciphertext using the symmetric key; and.....	43
o. Element 1[j]: (j) storing at least the key K in the embedded universal integrated circuit card for use in future communications.	44
3. Dependent Claims 2-17, 19-22	44
a. Claim 2: The method of claim 1, wherein the subscription manager public key is downloaded via the network application.	44
b. Claim 3: The method of claim 1, wherein the step of storing the subscription manager public key occurs at the time of manufacture of the embedded universal integrated circuit card.	45
c. Claim 4: The method of claim 1, wherein the key exchange algorithm is a Diffie-Hellman key exchange.	47
d. Claim 6: The method of claim 1, wherein the encrypted profile is encrypted in a manner so that intermediate nodes on an IP network would not be able to read data within the profile in an unencrypted form.....	47
e. Claim 7: The method of claim 1, wherein subscription manager system stores the ciphertext and the symmetric key before step (a).....	48

TABLE OF CONTENTS

(continued)

Page

f.	Claim 8: The method of claim 1, wherein after the step (b), the embedded universal integrated circuit card is authenticated by the subscription manager system using at least the identity of the embedded universal integrated circuit card included in the first message.	50
g.	Claim 10: The method of claim 1, wherein the decrypted ciphertext is stored in nonvolatile memory of the embedded universal integrated circuit card.	52
h.	Claim 11: The method of claim 1, wherein the embedded universal integrated circuit card is operably connected to a module and wherein the module is a machine-to-machine module.....	52
i.	Claim 12: The method of claim 11, wherein the module comprises a module of a type selected from the group consisting of: vending machine, automobile, alarm system, tracking device, monitoring device, security system and remote control device.....	53
j.	Claim 13: The method of claim 1, wherein the embedded universal integrated circuit card is operably connected to a module and wherein the module is a wireless module.....	53

TABLE OF CONTENTS

(continued)

Page

k.	Claim 15: The method of claim 13, wherein the module comprises a module of a type selected from the group consisting of the following: wireless handset, cellular phone, smartphone, laptop computer, tracking device, computer with a radio, circuit board with a radio, hand-held device, netbook, portable computer, multiprocessor system, microprocessor based consumer electronic, programmable consumer electronic, network personal computer, minicomputer, mainframe computer and smartcard.....	54
l.	Claim 16: The method of claim 1, wherein the embedded universal integrated circuit card is operably connected to a module, wherein the module is a wired module.....	55
m.	Claim 19: The method of claim 1, wherein the deriving step (c) is performed prior to the generating step (a).....	56
n.	Claim 21: The method of claim 1, wherein the embedded integrated circuit card is operably connected to a module and a user accesses a user interface of the module to enter authentication information before the module receives the symmetric key.....	57
o.	Claim 22: The method of claim 1, wherein the subscription manager receives the ciphertext from a mobile network operator before the network application sends the first message.....	58
B.	Ground 2: Claims 1-17 and 19-22 are obvious over Park in view of GlobalPlatform and X9.63-Overview.....	58
1.	A POSITA would have been motivated to combine Park's teachings with GlobalPlatform's and X9.63-Overview's teachings and would have had a reasonable expectation of success.....	58

TABLE OF CONTENTS

(continued)

	Page
2. Independent Claim 1	60
a. Elements 1[pre]-1[b] and 1[f]-1[j].....	60
b. Element 1[c]: (c) deriving a second eUICC private key and a corresponding second eUICC public key using a first random number generator and a first set of cryptographic algorithms;.....	60
c. Element 1[d]: (d) storing a subscription manager public key which corresponds to a subscription manager private key;	62
d. Element 1[e][1]: (e) deriving a profile key using a key exchange algorithm based on at least:	62
e. Element 1[e][2]: wherein the profile key can also be derived at the subscription manager system based at least on: the second eUICC public key, and the subscription manager private key;	63
3. Dependent Claims 2-17, 19-22	64
a. Claims 2-3, 6-17, 19-22	64
b. Claim 4: The method of claim 1, wherein the key exchange algorithm is a Diffie-Hellman key exchange.	64
C. Grounds 3-4: Claim 18 is obvious over Park in view of GlobalPlatform, AbiChar, and Weiss, and Park in view of GlobalPlatform, X9.63-Overview, and Weiss	65
1. A POSITA would have been motivated to add Weiss's teachings to the Park-GlobalPlatform-AbiChar and Park-GlobalPlatform-X9.63-Overview systems and would have had a reasonable expectation of success.	65
2. Claim 18: The method of claim 16, wherein the module comprises a module of a type selected from the group consisting of: computer, security camera, security monitoring device and networked controller.	65

TABLE OF CONTENTS

(continued)

	Page
D. Ground 5: Claims 1-22 are obvious over Nix175 in view of Park and GlobalPlatform	66
1. A POSITA would have been motivated to combine Nix175, Park, and GlobalPlatform and would have had a reasonable expectation of success.....	66
2. Independent Claim 1	71
a. Element 1[pre]: A method comprising:.....	71
b. Element 1[a][1]: (a) generating a first message comprising (1) an identity of an embedded universal integrated circuit card (eUICC); (2) a nonce; and (3) a first digital signature, generated using a first eUICC private key, wherein the first eUICC private key corresponds to a first eUICC public key;	72
c. Element 1[b]: (b) sending the first message via a network application to a subscription manager system;	73
d. Element 1[c]: (c) deriving a second eUICC private key and a corresponding second eUICC public key using a first random number generator and a first set of cryptographic algorithms;	74
e. Element 1[d]: (d) storing a subscription manager public key which corresponds to a subscription manager private key;	75
f. Element 1[e][1]: (e) deriving a profile key using a key exchange algorithm based on at least:	76
g. Element 1[e][2]: wherein the profile key can also be derived at the subscription manager system based at least on: the second eUICC public key, and the subscription manager private key;	77

TABLE OF CONTENTS

(continued)

	Page
h. Element 1[f]: (f) receiving from the subscription manager system, an encrypted profile comprising a ciphertext including a key K encrypted with a symmetric key;.....	77
i. Element 1[g]: (g) receiving the symmetric key;.....	78
j. Element 1[h]: (h) decrypting, by the embedded universal integrated circuit card, at least a portion of the encrypted profile using the profile key;	81
k. Element 1[i]: (i) decrypting, by the embedded universal integrated circuit card, at least a portion of the ciphertext using the symmetric key; and.....	82
l. Element 1[j]: (j) storing at least the key K in the embedded universal integrated circuit card for use in future communications.	83
3. Dependent Claims 2-22.....	83
a. Claim 2: The method of claim 1, wherein the subscription manager public key is downloaded via the network application.	83
a. Claim 3: The method of claim 1, wherein the step of storing the subscription manager public key occurs at the time of manufacture of the embedded universal integrated circuit card.	84
b. Claim 4: The method of claim 1, wherein the key exchange algorithm is a Diffie-Hellman key exchange.	85
c. Claim 6: The method of claim 1, wherein the encrypted profile is encrypted in a manner so that intermediate nodes on an IP network would not be able to read data within the profile in an unencrypted form.....	85
d. Claim 7: The method of claim 1, wherein subscription manager system stores the ciphertext and the symmetric key before step (a).....	86

TABLE OF CONTENTS
(continued)

	Page
e. Claim 8: The method of claim 1, wherein after the step (b), the embedded universal integrated circuit card is authenticated by the subscription manager system using at least the identity of the embedded universal integrated circuit card included in the first message.	87
f. Claim 10: The method of claim 1, wherein the decrypted ciphertext is stored in nonvolatile memory of the embedded universal integrated circuit card.	87
g. Claim 11: The method of claim 1, wherein the embedded universal integrated circuit card is operably connected to a module and wherein the module is a machine-to-machine module.....	88
h. Claim 12: The method of claim 11, wherein the module comprises a module of a type selected from the group consisting of: vending machine, automobile, alarm system, tracking device, monitoring device, security system and remote control device.....	88
i. Claim 13: The method of claim 1, wherein the embedded universal integrated circuit card is operably connected to a module and wherein the module is a wireless module.....	89

TABLE OF CONTENTS

(continued)

Page

j.	Claim 15: The method of claim 13, wherein the module comprises a module of a type selected from the group consisting of the following: wireless handset, cellular phone, smartphone, laptop computer, tracking device, computer with a radio, circuit board with a radio, hand-held device, netbook, portable computer, multiprocessor system, microprocessor based consumer electronic, programmable consumer electronic, network personal computer, minicomputer, mainframe computer and smartcard.....	90
k.	Claim 16: The method of claim 1, wherein the embedded universal integrated circuit card is operably connected to a module, wherein the module is a wired module.....	90
l.	Claim 18: The method of claim 16, wherein the module comprises a module of a type selected from the group consisting of: computer, security camera, security monitoring device and networked controller.....	91
m.	Claim 19: The method of claim 1, wherein the deriving step (c) is performed prior to the generating step (a).....	91
	Claim 20: The method of claim 1, wherein the generating step (a) is performed before the deriving step (c).	91
n.	Claim 21: The method of claim 1, wherein the embedded integrated circuit card is operably connected to a module and a user accesses a user interface of the module to enter authentication information before the module receives the symmetric key.....	92

TABLE OF CONTENTS
(continued)

	Page
o. Claim 22: The method of claim 1, wherein the subscription manager receives the ciphertext from a mobile network operator before the network application sends the first message.....	92
XIII. CONCLUSION.....	93

LIST OF EXHIBITS¹

Ex. No.	Description
Ex-1001	U.S. Patent No. 12,207,094 (“the ’094 Patent”)
Ex-1002	Declaration of Dr. Paul S. Min
Ex-1003	Curriculum Vitae of Dr. Paul S. Min
Ex-1004	Prosecution History of U.S. Patent No. 12,207,094
Ex-1005	Jaemin Park et al., Secure Profile Provisioning Architecture for Embedded UICC, Int’l Conference on Availability, Reliability & Security 297 (2013) (“Park”)
Ex-1006	GlobalPlatform Card Specification, Version 2.2.1, Public Release (Jan. 2011) (“GlobalPlatform”)
Ex-1007	Pierre E. Abi-Char et al., A Fast and Secure Elliptic Curve Based Authenticated Key Agreement Protocol For Low Power Mobile Communications, Int’l Conference on Next Generation Mobile Applications, Services and Technologies (2007) (“AbiChar”)
Ex-1008	ANSI X9.63 Overview, Key Agreement and Key Transport Using Elliptic Curve Cryptography, Simon Blake-Wilson, Certicom (2000) (“X9.63-Overview”)
Ex-1009	U.S. Patent Pub. No. 2015/0350881 to Weiss (“Weiss”)
Ex-1010	INTENTIONALLY LEFT BLANK
Ex-1011	GlobalPlatform Card Secure Channel Protocol ’11’ Card Specification v2.2 – Amendment F, Version 1.0 (May 2015) (“SCP11”)

¹ Four-digit pin citations that begin with 0 are to the branded numbers added by Samsung in the bottom right corner of the exhibits. All other pin citations are to original page, column, paragraph, or line numbers.

Ex. No.	Description
Ex-1012	INTENTIONALLY LEFT BLANK
Ex-1013	INTENTIONALLY LEFT BLANK
Ex-1014	INTENTIONALLY LEFT BLANK
Ex-1015	Declaration of Simon Blake-Wilson, author of ANSI X9.63-Overview
Ex-1016	U.S. Patent No. 9,100,175 (“Nix175”)
Ex-1017	INTENTIONALLY LEFT BLANK
Ex-1018	Declaration of Tono Aspinall of GlobalPlatform, Inc.
Ex-1019	Declaration of Gordan MacPherson of IEEE
Ex-1020	Claim Mapping Table
Ex-1021	Colin Boyd and Anish Mathuria, “Protocols for Authentication and Key Establishment” (Springer, 2003) (“Boyd & Mathuria”)
Ex-1022	Diffie and Hellman, “New Directions in Cryptography” IEEE Transactions on Information Theory, Vol. 22, No. 6 (November 1976)
Ex-1023	NIST Special Publication 800-57, Recommendation for Key Management – Part 1: General (Rev. 3) (July 2012)
Ex-1024	NIST Special Publication 800-38F, Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping (December 2012)
Ex-1025	NIST Special Publication 800-56A, Revision 2, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography (May 2013)
Ex-1026	GlobalPlatform Card Technology, Secure Channel Protocol 03, Card Specification v 2.2 – Amendment D, Version 1.0 (April 2009)

Ex. No.	Description
Ex-1027	Reprogrammable SIMs: Technology, Evolution and Implications, Final Report, Ofcom (September 25, 2012)
Ex-1028	ETSI TS 103 383 v12.1.0, “Smart Cards; Embedded UICC; Requirements Specification (Release 12),” (June 2013)
Ex-1029	Network-1’s Infringement Contentions for the ’094 Patent, filed on September 30, 2025, as Exhibit F to Network-1 Technologies, Inc.’s P.R. 3-1 Disclosure of Asserted Claims and Infringement Contentions
Ex-1030	U.S. Patent No. 11,916,893

I. INTRODUCTION

Samsung Electronics Co., Ltd. and Samsung Electronics America, Inc. (collectively, “Petitioner”) request *inter partes* review (“IPR”) of Claims 1-22 of U.S. Patent No. 12,207,094 (“the ’094 Patent”) (Ex-1001), currently assigned to Network-1 Technologies, Inc. (“PO”).

II. MANDATORY NOTICES UNDER 37 C.F.R. §42.8

Real Parties-in-Interest: Petitioner identifies the following real parties-in-interest: Samsung Electronics Co., Ltd. and Samsung Electronics America, Inc.

Related Matters: PO has asserted the ’094 Patent against Petitioner in *Network-1 Technologies, Inc. v. Samsung Electronics Co., Ltd. et al.*, No. 2:25-cv-00667 (E.D. Tex.) (“Samsung litigation”).

Lead and Back-Up Counsel:

- Lead Counsel:

William M. Fink (Reg. No. 72,332)
O’Melveny & Myers LLP
1625 Eye Street, NW
Washington, DC 20006
Telephone: (202) 383-5300
Fax: (202) 383-5414
E-Mail: tfink@omm.com

- Backup Counsel:

Benjamin M. Haber (Reg. No. 67,129)
O’Melveny & Myers LLP
400 South Hope Street, 18th Floor
Los Angeles, CA 90071

Telephone: (213) 430-6000
Fax: (213) 430-6407
Email: bhaber@omm.com

Marc J. Pensabene (Reg. No. 37,416)
O'Melveny & Myers LLP
1301 Avenue of the Americas, Suite 1700
New York, NY 10019
Telephone: (212) 326-2000
Fax: (212) 326-2061
Email: mpensabene@omm.com

Brian Cook (Reg. No. 59,356)
O'Melveny & Myers LLP
400 South Hope Street, 18th Floor
Los Angeles, CA 90071
Telephone: (213) 430-6000
Fax: (213) 430-6407
Email: bcook@omm.com

Caitlin P. Hogan (Reg. No. 61,515)
O'Melveny & Myers LLP
1301 Avenue of the Americas, Suite 1700
New York, NY 10019
Telephone: (212) 326-2000
Fax: (212) 326-2061
Email: chogan@omm.com

Service Information: Petitioner consents to electronic service by email to the following addresses:

- tfink@omm.com
- bhaber@omm.com
- mpensabene@omm.com
- bcook@omm.com
- chogan@omm.com

III. FEE AUTHORIZATION

The PTO is authorized to charge any fees due during this proceeding to Deposit Account No. LA500639.

IV. GROUNDS FOR STANDING

Petitioner certifies that the '094 Patent is available for review, and Petitioner is not barred or estopped from requesting review.

V. PRECISE RELIEF REQUESTED

Petitioner requests review and cancellation of Claims 1-22 as unpatentable based on the following grounds, supported by a declaration from Dr. Paul S. Min. Ex-1002 ¶¶1-83; Ex-1003.

Ground	Summary
1	Claims 1-17, and 19-22 are obvious over Park (Ex-1005) in view of GlobalPlatform (Ex-1006) and AbiChar (Ex-1007)
2	Claims 1-17, and 19-22 are obvious over Park in view of GlobalPlatform and X9.63-Overview (Ex-1008)
3-4	Claim 18 is obvious over Park in view of GlobalPlatform, AbiChar, and Weiss (Ex-1009), and Park in view of GlobalPlatform, X9.63-Overview, and Weiss
5	Claims 1-22 are obvious over Nix175 in view of Park and GlobalPlatform

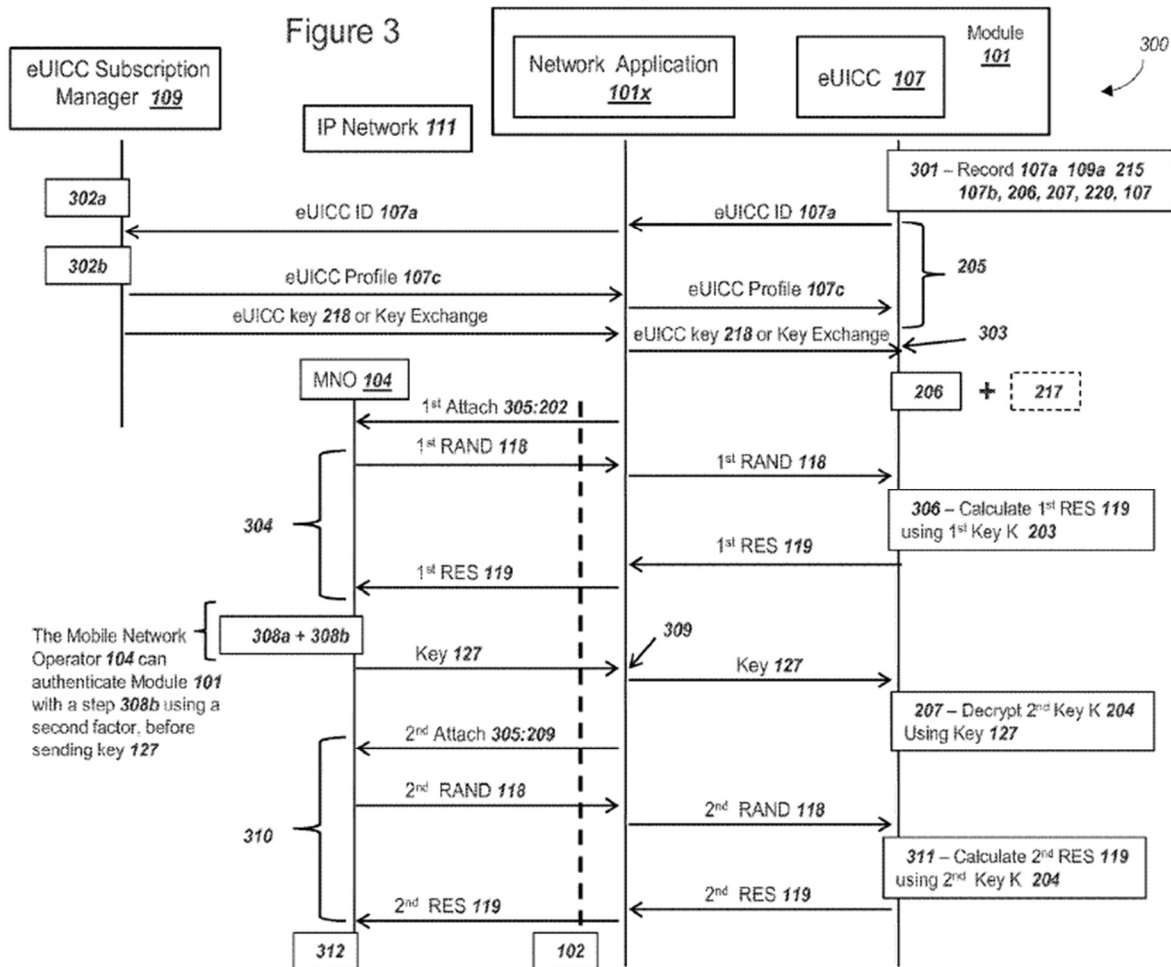
VI. THE CHALLENGED PATENT

The '094 Patent relates to techniques for provisioning an embedded universal integrated circuit card (eUICC) with a profile containing network access credentials. Ex-1001, 3:62-4:4.

As the '094 Patent explains, mobile devices traditionally relied on a “subscriber identity module (SIM) card” or “a related universal integrated circuit card (UICC)” to access a wireless network. Ex-1001, 2:41-52. Each card was supplied by a mobile network operator (MNO) and contained “a pre-shared secret key K in addition to a set of parameters for a mobile phone or user equipment to connect with the wireless network operated by the MNO.” Ex-1001, 2:52-56. Because each card only allowed access to the network of its associated MNO, changing networks required “swapping the UICC card,” the cost of which “could be prohibitive.” Ex-1001, 3:8-3:22. This led the industry “to consider an embedded UICC” (eUICC), onto which new MNO network access credentials could be transferred, without physically swapping out the UICC. Ex-1001, 3:23-27.

The '094 Patent purports to address a security concern with this new provisioning system, where the distribution and release of the profile (including decryption of a secret key K) is under the control of the SM—and not the MNO, which thus loses control over ensuring its secret key K is only provided to authorized devices. Ex-1001, 3:66-4:4, 35:45-56. To address this issue, the '780 Patent

describes a multi-stage provisioning process illustrated in Figure 3, below, splitting control between the SM (for delivering the secure profile) and the MNO (for transmitting the symmetric key used to encrypt secret key K).²



The profile 107c delivered to the eUICC by the SM includes two portions—a first portion including a 1st Key K, and a second portion including a 2nd Key K that is encrypted under a symmetric key. *Id.*, 5:42-49, 6:47-49. The eUICC receives this

² Throughout, emphasis and annotations are added unless otherwise specified.

profile from the SM, encrypted under a “profile key” that is known to both the SM and the eUICC. *Id.*, 5:41-47. After decrypting the first portion of the profile using its stored profile key, the eUICC can perform a two-factor authentication with the MNO. *Id.*, 5:52-6:2; Fig. 3, 308a, 308b. After a second-factor authentication, the MNO transmits the symmetric key needed to decrypt the second portion of the profile. *Id.*, 6:37-39. Using this symmetric key, the eUICC completes the profile decryption, thereby recovering the second key K, which it can use to gain full access to the network. *Id.*, 6:50-7:9.

Thus, the purpose of the '094 Patent is its two-stage provisioning process, which ensures that the network operator—not the subscription manager—is in control of the final release of network credentials. However, Claim 1 (reproduced below) is drafted quite differently, reciting an eUICC performing three conventional processes: (1) Elements (a)-(b) recite sending a profile provisioning request with information that can be used to authenticate the eUICC and using a first eUICC private key to create a digital signature; (2) Elements (c)-(e) recite participating in an asymmetric key exchange using a second derived eUICC private/public key pair to generate a shared “profile key” for use in encrypting/decrypting the profile; and (3) Elements (f)-(j) recite receiving and decrypting the encrypted profile, where the profile is encrypted by two separate keys—the profile key encrypting a first portion

of the profile, and a symmetric key encrypting a second portion of the profile that includes a network access key K. Ex-1002 ¶¶84-89.

1. A method comprising:

- (a) generating a first message comprising:
 - (1) an identity of an embedded universal integrated circuit card (eUICC);
 - (2) a nonce; and
 - (3) a first digital signature, generated using a first eUICC private key, wherein the first eUICC private key corresponds to a first eUICC public key;
- (b) sending the first message via a network application to a subscription manager system;

profile provision request with information needed to authenticate the eUICC

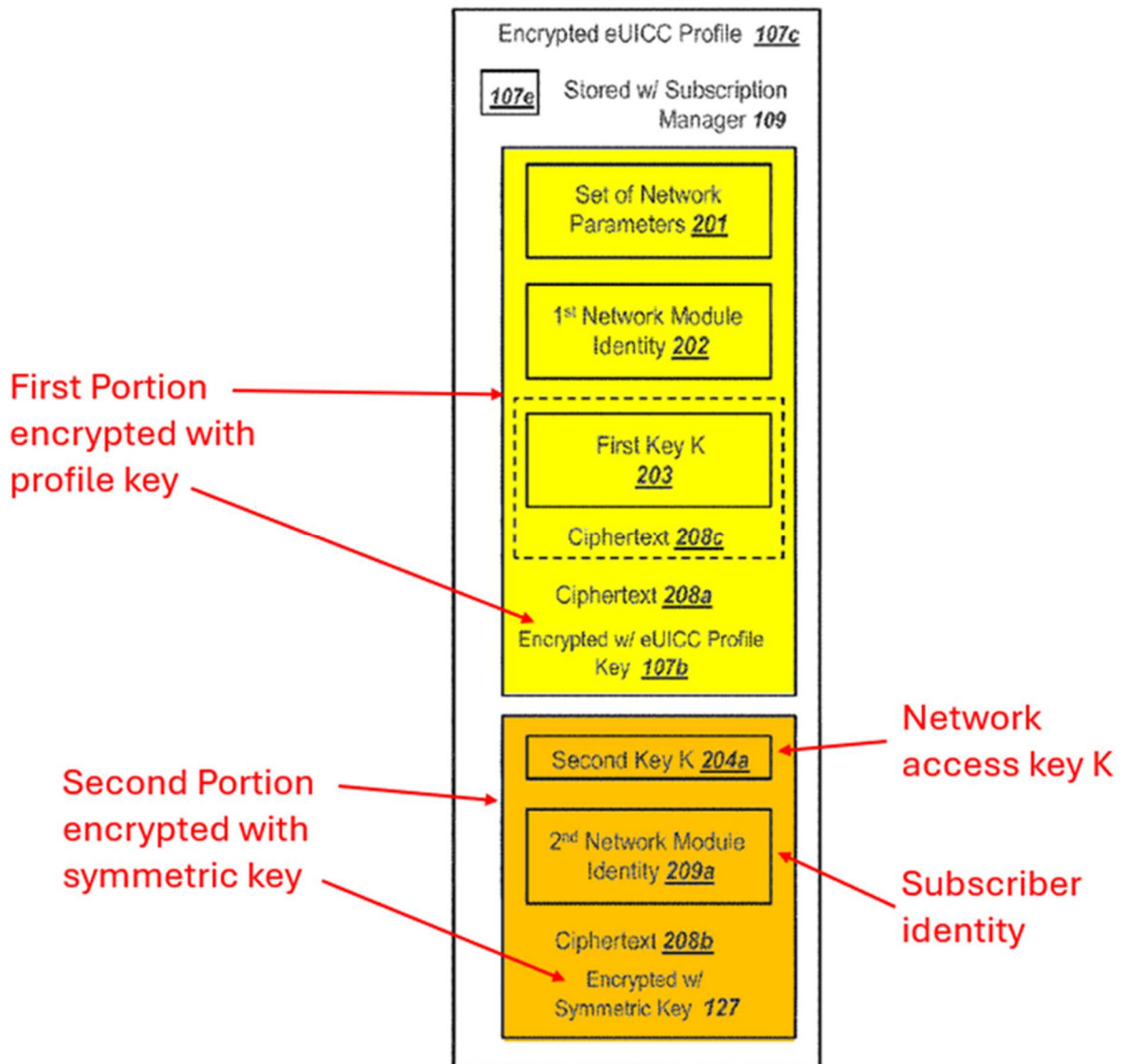
- (c) deriving a second eUICC private key and a corresponding second eUICC public key using a first random number generator and a first set of cryptographic algorithms;
- (d) storing a subscription manager public key which corresponds to a subscription manager private key;
- (e) deriving a profile key using a key exchange algorithm based on at least:
 - (i) the second eUICC private key, and
 - (ii) the stored subscription manager public key, wherein the profile key can also be derived at the subscription manager system based at least on: the second eUICC public key, and the subscription manager private key;

perform an asymmetric key exchange with an SM to derive a shared, symmetric "profile key"

- (f) receiving from the subscription manager system, an encrypted profile comprising a ciphertext including a key K encrypted with a symmetric key;
- (g) receiving the symmetric key;
- (h) decrypting, by the embedded universal integrated circuit card, at least a portion of the encrypted profile using the profile key;
- (i) decrypting, by the embedded universal integrated circuit card, at least a portion of the ciphertext using the symmetric key; and
- (j) storing at least the key K in the embedded universal integrated circuit card for use in future communications.

receive encrypted profile having an encrypted, second portion, that contains secret key K, and is decrypted using a separately received symmetric key

The annotated excerpt of Figure 2a, below, illustrates the two portions of the profile encrypted by two different keys:



VII. PROSECUTION HISTORY OF THE '094 PATENT

The '094 Patent was filed January 18, 2024, and claims priority through a chain of 6 continuation applications (now issued patents), the earliest of which was filed December 6, 2013. During prosecution, the Examiner did not issue any substantive prior art rejections, only double patenting and Section 112 rejections,

which the Applicant overcame by filing terminal disclaimers and amending the claims to provide antecedent basis for the claimed eUICC. Ex-1004; Ex-1002 ¶¶90.

VIII. LEVEL OF ORDINARY SKILL IN THE ART

A person of ordinary skill in the art at the relevant time (“POSITA”) would have had at least a bachelor’s degree in electrical engineering, computer engineering, computer science, or a similar field, and 2-3 years of experience with cellular/WLAN security and mobile devices. Ex-1002 ¶¶25-27.

IX. PRIORITY DATE

The ’094 Patent claims a priority date of December 6, 2013. However, the claims of the ’094 Patent are not supported by the written disclosure of the earliest priority application for two reasons. Ex-1002 ¶¶91-94.

First, while element 1(a)(3) recites a “first eUICC public key” and element 1(c) recites a “second eUICC public key,” the specification discloses only one eUICC private/public key pair: eUICC private key 215 and eUICC public key 214 and does not disclose two separate pairs. *E.g.*, Ex-1001, 43:36-67; Ex-1016, 42:11-42.

Second, the common specification of the ’094 Patent and its predecessors make clear that it is the MNO that sends the symmetric key to the eUICC, to ensure that the MNO maintains control over the release of its network access key K. *See, e.g.*, Ex-1001, 6:37-39, 35:19-45, 35:56-62, 41:65-42:13, 59:44-55, 59:62-60:6.

However, the claims of the '094 Patent place no limit on the source of the claimed “symmetric key,” and, in the parties’ district court litigation, PO has taken the position that the eUICC receives the symmetric key not from the MNO, but from the subscription manager. Ex-1029 (ICs); Ex-1002 ¶¶92.

That the symmetric key may be sent from the subscription manner, rather than the MNO, was first disclosed within the '094 Patent’s priority chain in the claims of the '893 Patent. Ex-1030, Cl.1; Ex-1002 ¶¶93. Accordingly, the '094 Patent is entitled to a priority date no earlier than the filing date of the '893 Patent: December 10, 2021. *See, e.g., X2Y Attenuators, LLC v. ITC*, 757 F.3d 1358, 1365 (Fed. Cir. 2014) (“[A] claim is entitled to the priority date of an earlier application only if the earlier specification provides sufficient written support for the full scope of the claim.”).

Ground 5 assumes this later priority date, while the references relied upon in Grounds 1-4 are prior art under either priority date.

X. CLAIM CONSTRUCTION

Petitioner interprets the claims according to *Phillips*. 37 C.F.R. §42.100(b). To resolve this Petition, Petitioner does not believe that any term requires express construction.³ CTPG, 44; Ex-1002 ¶¶95.

³ The Samsung litigation is in its infancy and claim construction proceedings have only just begun. Petitioner respectfully reserves all rights to raise claim construction and other arguments in district court and will request leave to submit the district

XI. BRIEF DESCRIPTION OF THE APPLIED PRIOR ART

A. Park (Ex-1005)

Park was published in the Proceedings of the 2013 International Conference on Availability, Reliability and Security, Regensburg, Germany (297-303), cataloged by IEEE, and was publicly available by at least November 7, 2013. Park is prior art to the '094 Patent under 35 U.S.C. §102(a)(1). Ex-1019; Ex-1002 ¶¶96-98.

Park teaches a secure architecture for remotely provisioning MNO profiles onto eUICCs. Ex-1005, 297. Under ETSI/GSMA's new eUICC provisioning model (illustrated in Figure 1, below), the SM was divided into two roles: "SM-DP (Data Preparation)" for generating and encrypting the operator profile, and "SM-SR (Secure Routing)" for transporting and installing the encrypted profile onto the eUICC. *Id.*, 297.

court's claim construction as soon as it becomes available. Additionally, should PO raise express or implied claim construction arguments in its Preliminary Response, Petitioner will respectfully seek a Preliminary Reply to respond. Consolidated Trial Practice Guide (CTPG), 44-45.

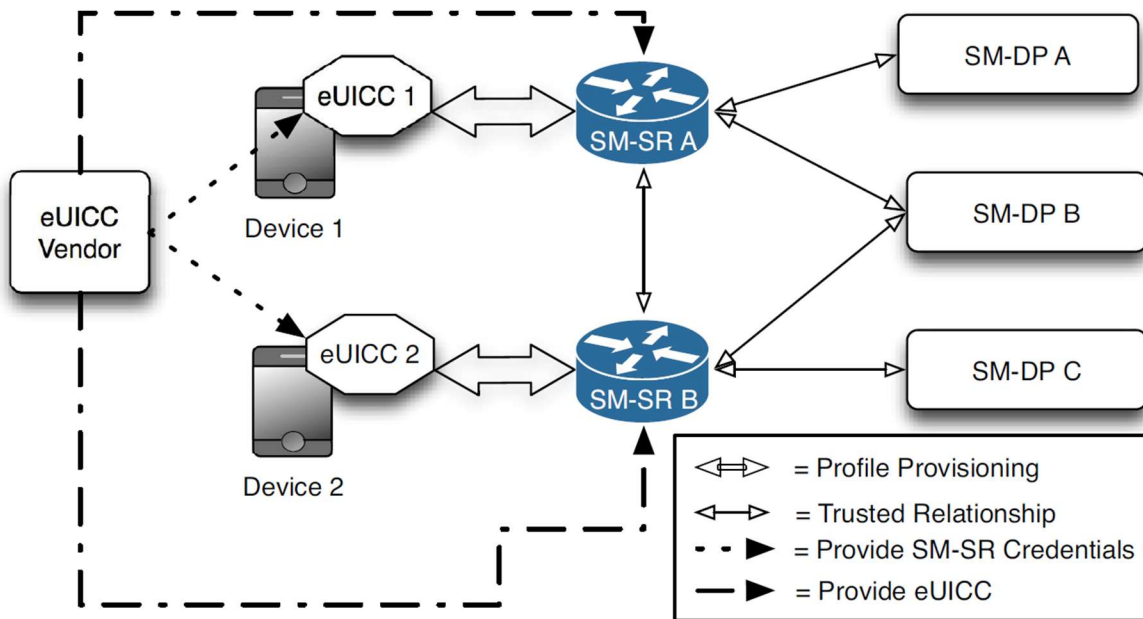


Fig. 1. eUICC Profile Ecosystem

This architecture introduced a new trust boundary—because the SM-SR may be operated by an entity distinct from the SM-DP or MNO, it may not be trusted with access to sensitive profile contents. *Id.* To mitigate, Park introduces a Key Agreement Module (KAM), enabling the eUICC and the SM-DP to negotiate mutually compatible profile encryption keys directly, without exposing them to the SM-SR or other intermediaries. *Id.*, 299-301.

B. GlobalPlatform (Ex-1006)

GlobalPlatform (GlobalPlatform Card Specification Version 2.2.1) was published in January 2011, made available that month on the GlobalPlatform Digital

Library web site, and is prior art to the '094 Patent under 35 U.S.C. §102(a)(1). Ex-1018; Ex-1002 ¶¶99-103.

GlobalPlatform defines a standardized architecture for smart card security including Secure Channel Protocols (SCPs) to protect communications between the card and an off-card entity, ensuring authentication, integrity, and confidentiality. Ex-1006, 19-20, 28, 112.

GlobalPlatform's Secure Channel Protocol 10 ("SCP10"), uses "a Public Key Infrastructure (PKI)" For authentication. *Id.*, 116. Under PKI, each participant has a pair of keys—a private key kept secret and a public key shared with others. Ex-1002 ¶101. During authentication, each party sends the other a random challenge, which the receiving party signs with its private key and returns to the sender. Ex-1006, 246-47, 266-68, 271. The sender then verifies that signature using the responder's public key. *Id.* Successful verification proves that the responder is the legitimate holder of the private key and therefore authentic, as shown in Figure F-9, below. *Id.*, 234.

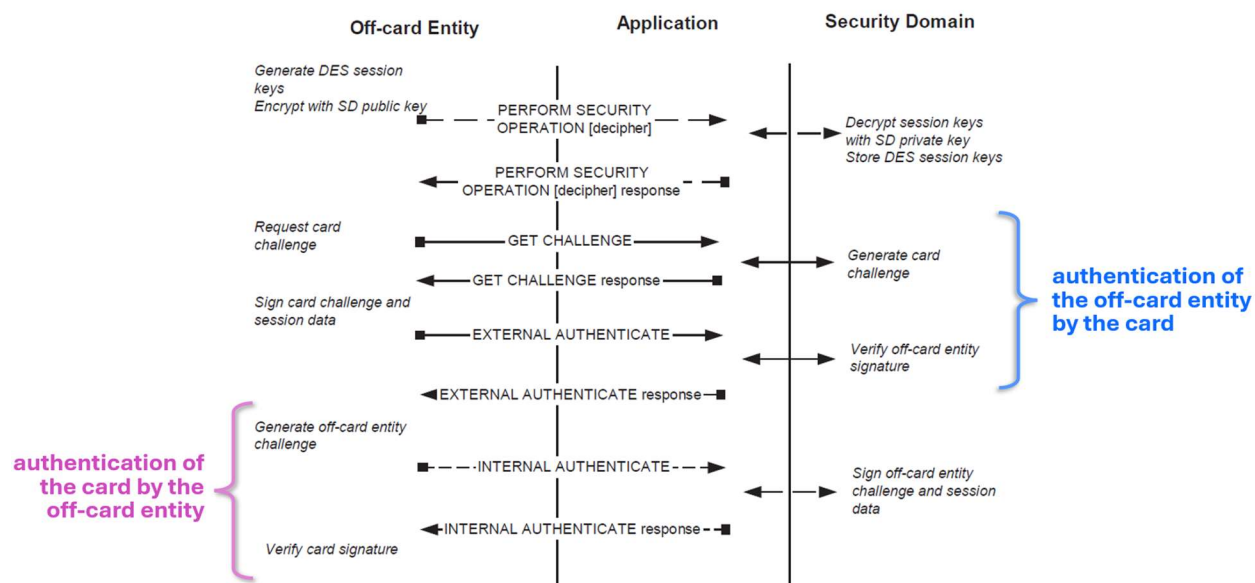


Figure F-9: Entity Authentication Flow

For confidentiality, a “Secure Channel session key” encrypts all message data transmitted to and from the card, as illustrated in Figure F-11. *Id.*, 258-59.

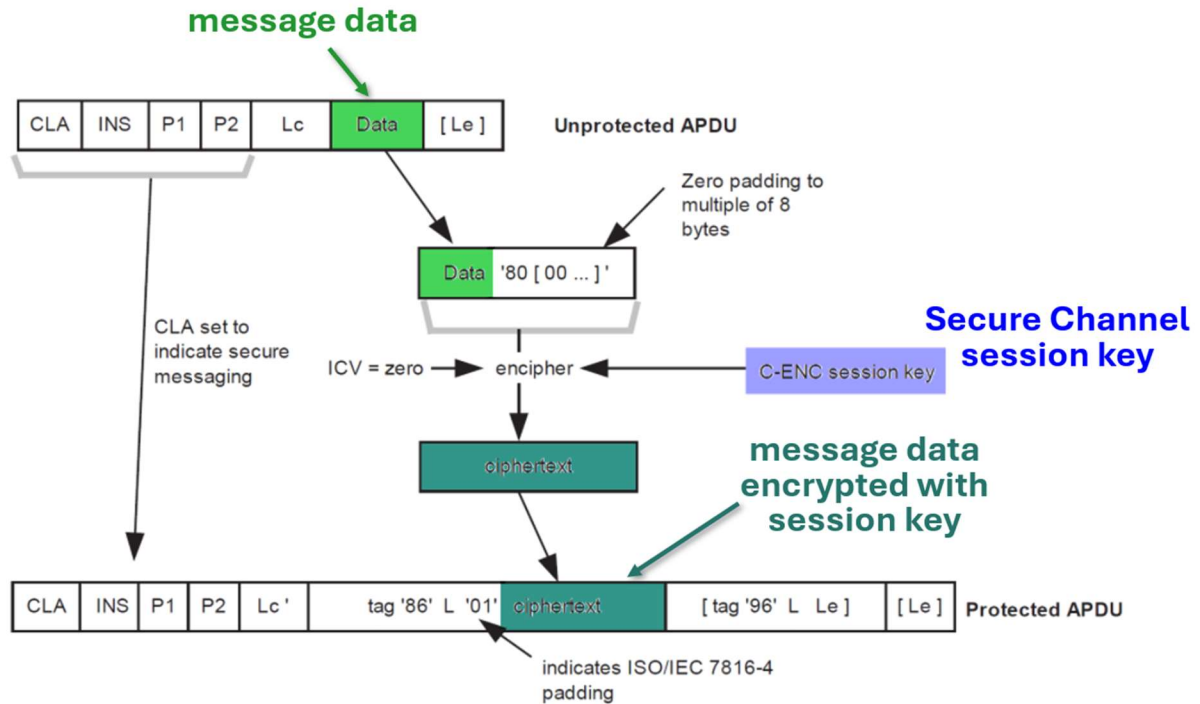


Figure F-11: Secure Messaging: Command message protected for confidentiality

GlobalPlatform imposes an additional level of protection for sensitive data, explaining that “[a] further level of security applies to sensitive data (e.g. secret keys) that shall always be transmitted as confidential data.” *Id.*, 112. This additional encryption “is over and beyond the Current Security Level required for the Secure Channel.” *Id.*, 264. Thus, when “transmitting sensitive data to and from the card”—including “all keys transmitted to a card”—the sensitive data is first encrypted “us[ing] the relevant data encryption session key (DEK)” and the entire message then further encrypted with the Secure Channel session key (C-ENC), providing layered protection for these highly sensitive materials. *Id.*, 264.

C. AbiChar (Ex-1007)

AbiChar was published in the Proceedings of the 2007 International Conference on Next Generation Mobile Applications, Services and Technologies (NGMAST 2007), was cataloged and made publicly available by IEEE by at least October 8, 2007, and is prior art to the '094 Patent under 35 U.S.C. §102(a)(1). Ex-1018; Ex-1002 ¶¶104-05.

AbiChar teaches a fast and secure elliptic-curve-based authenticated key agreement protocol (EC-SAKA) for low-power mobile communications. Ex-1007, 1. But as a starting point, AbiChar uses the standard “Diffie-Hellman key exchange based on EC,” (or ECDH), which corresponds to '780 claim 1(c)-1(e), and which AbiChar describes as follows:

Alice selects a random number n_a and computes $Y_a = n_a B$, she sends Y_a to Bob. Similarly Bob computes $Y_b = n_b B$ and sends Y_b to Alice. Alice and Bob generate the same key $K = n_a Y_b = n_b Y_a = n_a n_b B$.

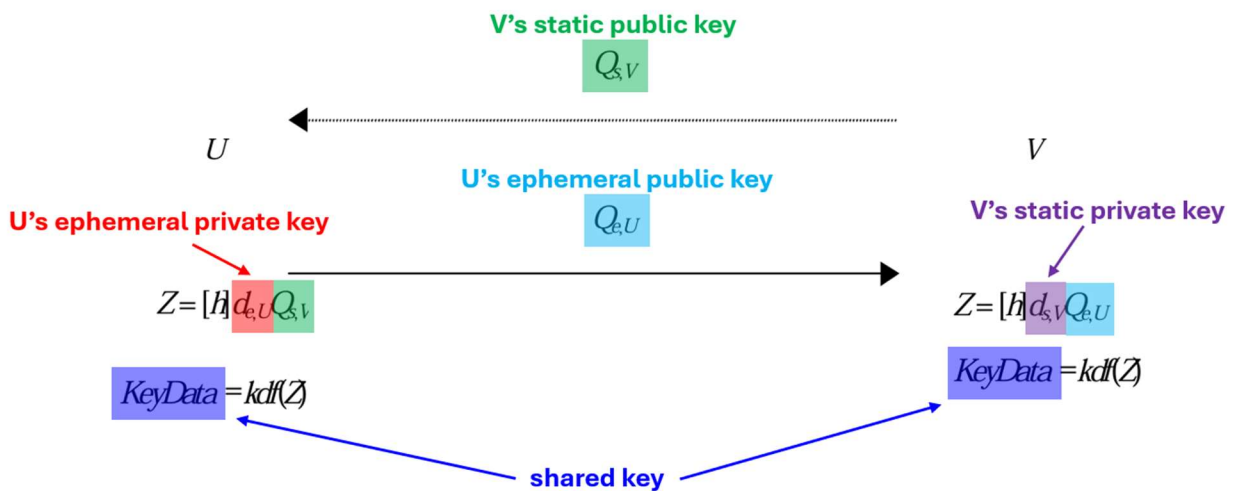
Ex-1007, 3. The derived key K in AbiChar corresponds to the profile key derived in claim 1(e).

D. X9.63-Overview (Ex-1008)

X9.63-Overview (ANSI X9.63 Overview, Key Agreement and Key Transport Using Elliptic Curve Cryptography) was authored by Simon Blake-Wilson in 1999, was publicly available in 2000, and is prior art to the '094 Patent under 35 U.S.C. §102(a)(1). Ex-1015; Ex-1002 ¶¶106-07.

X9.63-Overview “[s]pecifies key agreement and key transport schemes using elliptic curve cryptography” (Ex-1008, 3), including the one-pass Diffie-Hellman key-agreement protocol, illustrated below:

ANSI X9.63 - 1-Pass DH



Id., 12. Here, an entity (U) generates an ephemeral public/private key pair ($Q_{e,U}$, $d_{e,U}$) (*id.*), by randomly selecting an ephemeral private key ($d_{e,U}$) (*id.*, 7). Entity U then sends its ephemeral public key ($Q_{e,U}$) to entity (V). *Id.*, 12. Each entity then computes a shared secret Z using its own private key ($d_{e,U}$ for U, and $d_{s,V}$ for V) and the public key received from the other entity ($Q_{s,V}$ for U, and $Q_{e,U}$ for V). *Id.* The shared secret is then input into a key-derivation function (KDF) to produce a **shared key** (**KeyData**). *Id.* The resulting KeyData is substantially similar to AbiChar’s K and the “profile key” derived in claim 1(e).

E. Weiss (Ex-1009)

Weiss (U.S. Patent Publication No. 2015/0350881) filed on December 17, 2013, as PCT Application No. PCT/EP2013/003.813, claiming foreign application priority to December 21, 2012, and is prior art to the '094 Patent under 35 U.S.C. §102(a)(2). Ex-1002 ¶¶108-09.

Weiss teaches that “secure elements, such as SIMs, eUICCs, UICCs” (Ex-1009, ¶3) may be included in a variety of devices, “such as a tablet or notebook computer, a TV system, a set top box, a vending machine, a vehicle, a surveillance camera, a sensor device and the like” (*id.*, ¶31).

F. Nix175 (Ex-1016)

Nix175 (U.S. Patent No. 9,100,175) was filed on December 6, 2013 and published on August 4, 2015. If the challenged patent is not entitled to its priority claim as discussed in Section **Error! Reference source not found.**, Nix175 is prior art to the '094 Patent under 35 U.S.C. §102(a)(1). Ex-1016, cover; Ex-1002 ¶110. Nix175 shares a specification with the challenged patent.

XII. DETAILED EXPLANATION OF THE UNPATENTABILITY GROUNDS

A. Ground 1: Claims 1-17 and 19-22 are obvious over Park in view of GlobalPlatform and AbiChar.

1. **A POSITA would have been motivated to combine Park’s teachings with GlobalPlatform’s and AbiChar’s teachings and would have had a reasonable expectation of success.**

A POSITA would have been motivated to combine Park’s secure profile provisioning architecture (SPA) with GlobalPlatform’s procedures for mutual authentication and sensitive data encryption, and AbiChar’s elliptic curve Diffie-Hellman (ECDH) key agreement protocol, to yield a complete, secure, and efficient architecture for eUICC profile provisioning, as detailed below. Ex-1002 ¶111.

- a. **A POSITA would have turned to GlobalPlatform for implementation-level details of Park’s security requirements.**

Park’s Secure Profile Provisioning Architecture (SPA) defines an architectural framework for secure eUICC profile provisioning but intentionally leaves some implementation-level details to the practitioner. Ex-1005, 299-303. Park generates profiles and securely delivers them from the SM-DP to the eUICC, noting “diverse key agreement protocols, secure communication protocols and cryptographic algorithms can be applied depending on the circumstances around the eUICC.” *Id.*, 303. Moreover, Park recognizes the GlobalPlatform Card Specification (GP) and its Secure Channel Protocol (SCP) as the “de facto standard for profile provisioning,” and presents its SPA as extending GlobalPlatform. Ex-1005, 298. Park’s SPA’s fundamental security requirements—mutual authentication, and profile confidentiality—are derived directly from GlobalPlatform. Park explains that

SPA's security requirements "should be guaranteed in accordance with the requirements in [5]," where reference [5] corresponds to GlobalPlatform. *Id.*, 298, 303.

Thus, a POSITA would have looked to GlobalPlatform to understand how to implement Park's security requirements. Ex-1002 ¶¶112-13. GlobalPlatform's SCP's were well-known, widely deployed mechanisms for providing mutual authentication and confidentiality in smart-card provisioning. Ex-1002 ¶¶68-75, 113. Incorporating GlobalPlatform's authentication and encryption procedures into Park's architecture, would (1) directly satisfy Park's stated requirements, (2) align with the standard that Park's SPA builds upon, and (3) leverage mature, well-tested, and industry-accepted methods. Ex-1002 ¶113.

Mutual Authentication: The combined Park-GlobalPlatform system would achieve mutual authentication between the eUICC and the SM-DP using random challenges. Ex-1002 ¶114. GlobalPlatform's SCP10 performs PKI authentication whereby each side sends a random challenge to the other side, which then signs it with its private key. Ex-1006, 246-47, 266-68, 271; Ex-1002 ¶59. The sender then verifies that private key signature using the responder's public key, as illustrated in Figure F-9. *Id.*

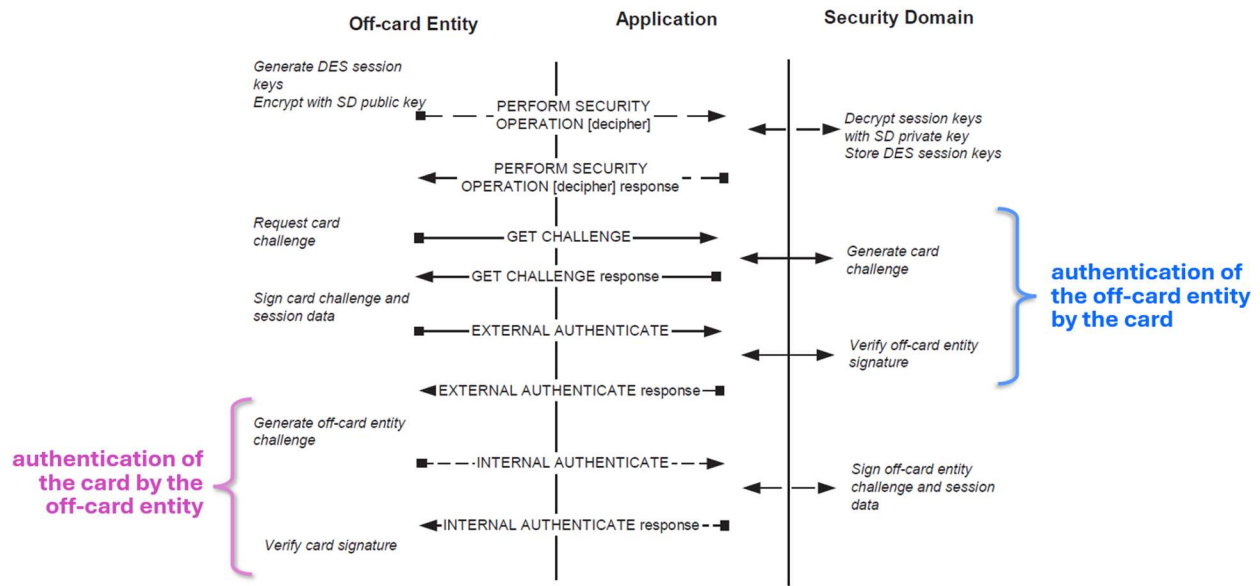
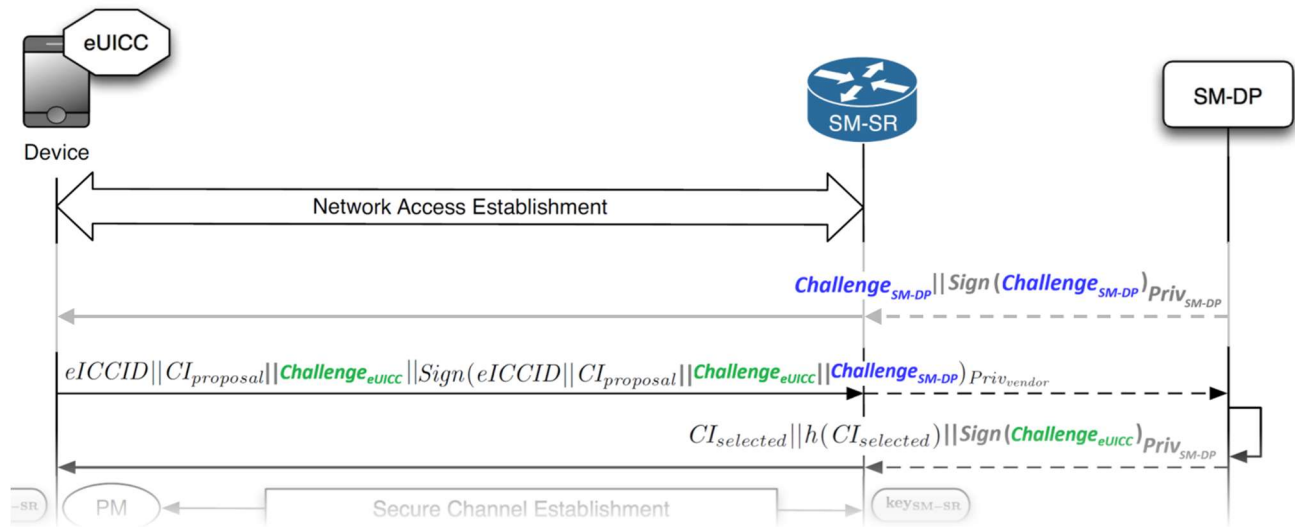


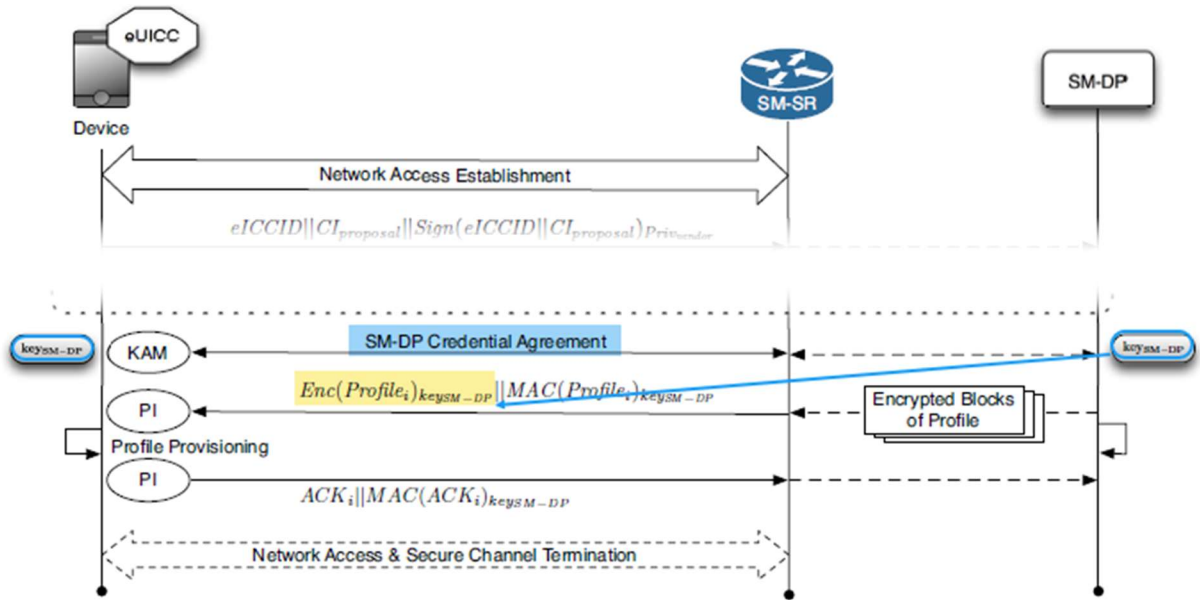
Figure F-9: Entity Authentication Flow

Given the limited processing and communication resources of eUICCs, a POSITA would have found it advantageous to include these random challenges and responses (to the extent possible) in existing messages sent between the eUICC to the SM-DP—such as Park’s capability negotiation message and the SM-DP’s response—rather than in a new communication. Ex-1002 ¶115. As illustrated in the modified version of Park’s Figure 4, below, it would have been a straightforward process to add these random challenges and responses to Park’s capability negotiation messages:



Ex-1002 ¶115. A POSITA would also have recognized that doing so would have had the added benefit of adding replay protection for the capability negotiation message, ensuring that an intercepted or previously transmitted message could not be reused by an attacker. Ex-1002 ¶¶60-67, 115.

Confidentiality: In Park's SPA, the eUICC's Key Agreement Module (KAM) is used to generate a shared SM-DP (below, blue) credential for encrypting and decrypting the profile (below, yellow) transmitted from the SM-DP to the eUICC. Ex-1005, 302. In the combined Park-GlobalPlatform system, this SM-DP credential would serve the same function as GlobalPlatform's Secure Channel session key—providing confidentiality for data exchanged between an off-card entity and the card. Ex-1005, 302; Ex-1006, 258-59; Ex-1002 ¶¶75, 116.



As Park explains, the SM-DP uses the SM-DP credential, together with “selected cryptographic algorithms” agreed upon by the parties during the capability negotiation process, to encrypt the profile that it sends to the eUICC. Ex-1005, 300. GlobalPlatform provides those established encryption procedures. Ex-1002 ¶117.

As explained in Section XII.A.1, when transmitting sensitive data, such as secret/private keys, SPC10 first encrypts the sensitive data “us[ing] the relevant data encryption session key (DEK)” and then further encrypts the message with the secure channel session key, providing layered protection for these highly sensitive materials. Ex-1006, 264, 258 (explaining that the session key, C-ENC, is used to encrypt “the command data field of the command message being transmitted to the card,” which “includes any data within the data field that has *already been protected* for another purpose *e.g. secret or private keys encrypted with the data encryption*

session key”). This configuration is illustrated below in annotated Figure F-11. Ex-1002 ¶118.

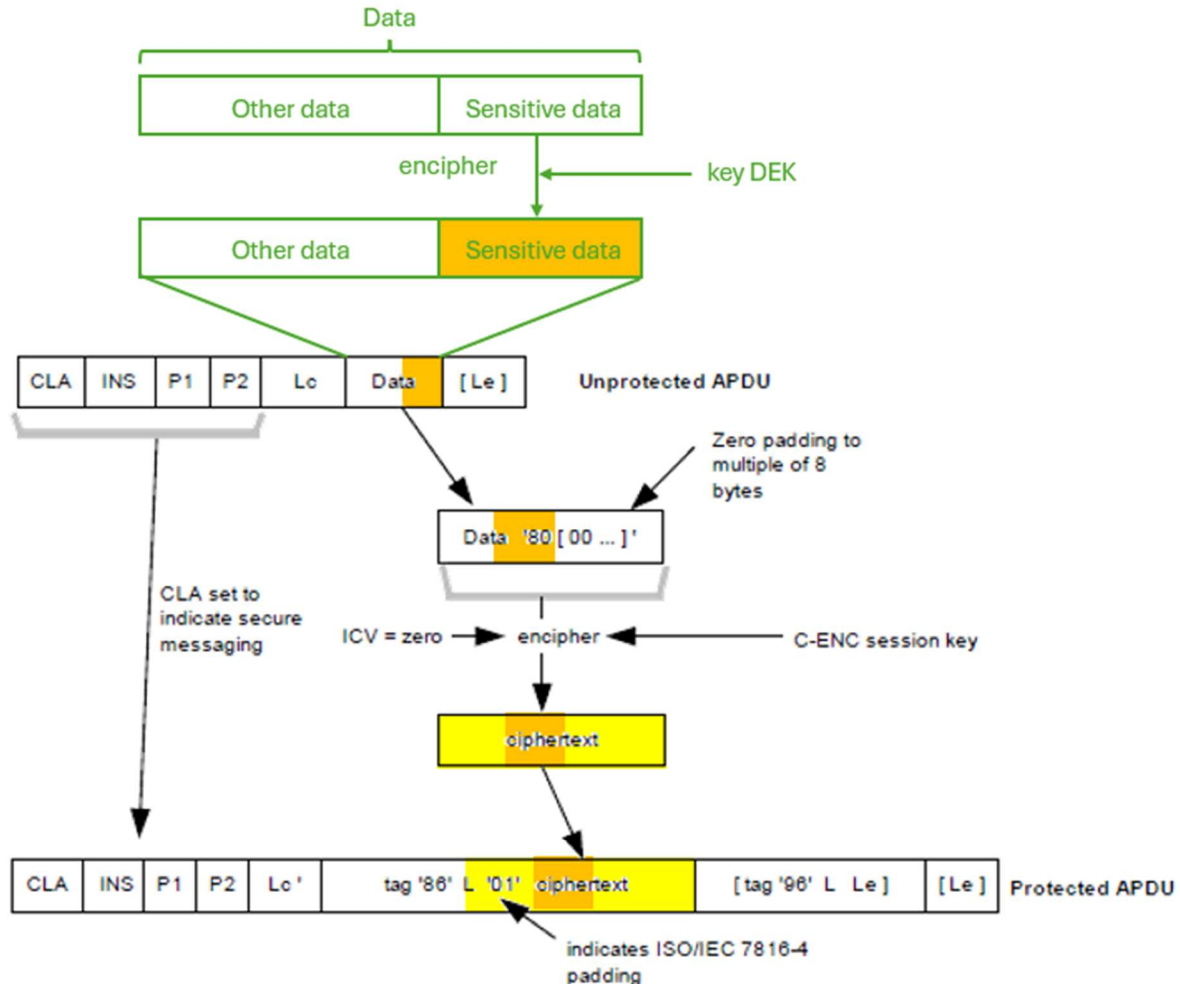


Figure F-11: Secure Messaging: Command message protected for confidentiality

Park similarly explains that eUICC profiles contain “security-sensitive information,” including “keys” used for accessing a mobile network. Ex-1005, 297. A POSITA would have recognized that applying GlobalPlatform’s DEK-based inner encryption to these elements of Park’s profile would provide an additional layer of

protection for the most sensitive data within the profile, ensuring that even if the outer session-key encryption were compromised, the critical key materials could remain protected. Ex-1002 ¶¶40-41, 55-56, 119. Thus, Park-GlobalPlatform sends profiles from the SM-DP to the eUICC that include an inner layer of encryption for sensitive materials like keys (using GlobalPlatform’s DEK), as well as an outer layer of encryption using Park’s key_{SM-DP} (corresponding to GlobalPlatform’s C-ENC session key). Ex-1002 ¶119.

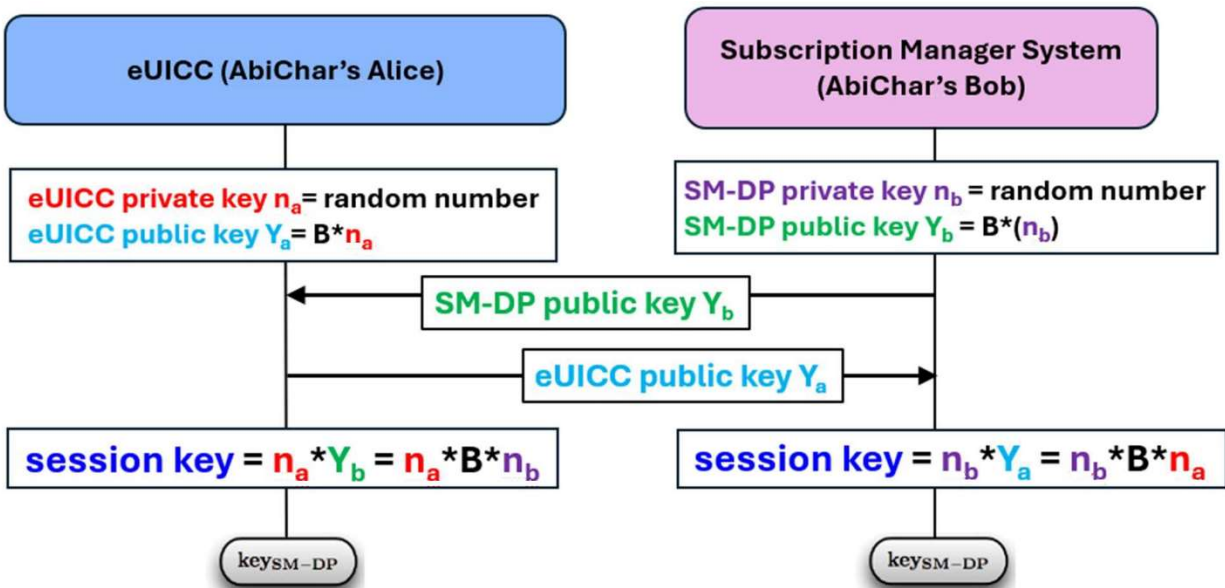
b. A POSITA would have understood that Abi-Char’s elliptic curve Diffie-Hellman key agreement could be implemented within Park’s SPA to provide an efficient asymmetric key-exchange mechanism

Park’s Key Agreement Module (KAM) “provides the *asymmetric algorithm-based key agreement protocol*” used to derive key_{SM-DP} for encrypting the eUICC profile. Ex-1005, 301. But Park does not specify a particular algorithm, instead explaining that “diverse key agreement protocols ... can be applied depending on the circumstances around the eUICC.” Ex-1005, 303.

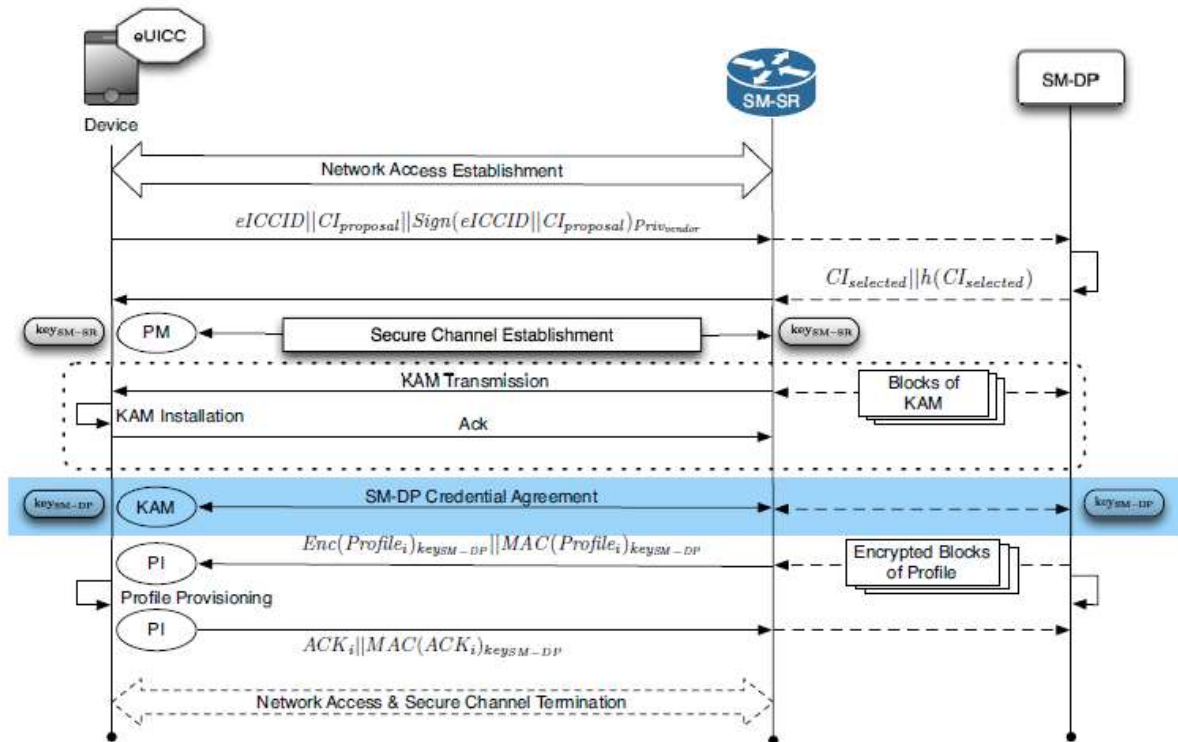
A POSITA would have recognized that the elliptic-curve-Diffie-Hellman key agreement protocol described in AbiChar provides such an efficient asymmetric key agreement technique suitable for Park’s architecture. Ex-1002 ¶¶42-48, 120-21. Abi-Char explains elliptic-curve cryptography “can significantly reduce the computation and storage overhead” of public-key operations, making it particularly well-suited

for mobile and smart-card devices with limited processing resources. Ex-1007, 1. AbiChar describes an elliptic-curve Diffie-Hellman (ECDH) key-agreement exchange, wherein each party generates random private keys (n_a and n_b respectively), generates public keys ($Y_a = n_a B$ and $Y_b = n_b B$, respectively), and then exchanges public keys. *Id.*, 3. Each party then calculates a shared key using its private key and the other party's public key ($K = n_a Y_b = n_b Y_a = n_a n_b B$). *Id.*

A POSITA would have incorporated AbiChar's ECDH protocol within Park's flexible, asymmetric framework to provide a secure, lightweight, and computationally efficient mechanism for establishing key_{SM-DP} between the eUICC and the SM-DP. Ex-1002 ¶¶45-48, 122. By mapping AbiChar's "Alice" to the eUICC and "Bob" to the SM-DP, the ECDH key exchange between the eUICC and SM-DP is illustrated as follows:



The ECDH exchange occurs in the Park-GlobalPlatform combination after the KAM transmission and installation (which would include ECDH parameters, such as the elliptic curve) and result in the establishment of keySM-DP in the “SM-DP Credential Agreement” step:



Ex-1002 ¶123.

2. Independent Claim 1

a. Element 1[pre]: A method comprising:

Park, GlobalPlatform, and AbiChar (“Park-GlobalPlatform-AbiChar”) teaches⁴ the preamble, to the extent it is limiting. Ex-1002 ¶¶124-25.

⁴ Petitioner uses the term “teaches” to include both express teachings and those fairly

Park teaches a method for provisioning a profile onto an “[e]mbedded UICC (hereafter eUICC),” using a “secure profile provisioning architecture.” Ex-1002, 297.

b. Element 1[a][1]: (a) generating a first message comprising

Park-GlobalPlatform-AbiChar teaches Element 1[a][1]. Ex-1002 ¶¶126-27.

Park teaches that the eUICC and SM-DP negotiate (via the SM-SR) to establish the key exchange protocol to be used by the parties. Ex-1005, 301. The eUICC sends a message to the SM-SR that includes “the eUICC ID (*eUICCID*) and its capability information (*CI_{proposal}*),” as illustrated in Figure 4, below. Ex-1005, 300. This capability negotiation message corresponds to the claimed “first message.”

suggested to a person of ordinary skill in the art. *In re Baird*, 16 F.3d 380, 383 (Fed. Cir. 1994); *In re Keller*, 642 F.2d 413, 425 (CCPA, 1981) (“The test for obviousness is...what the combined teachings of the references would have suggested to those of ordinary skill in the art.” (citations omitted)).

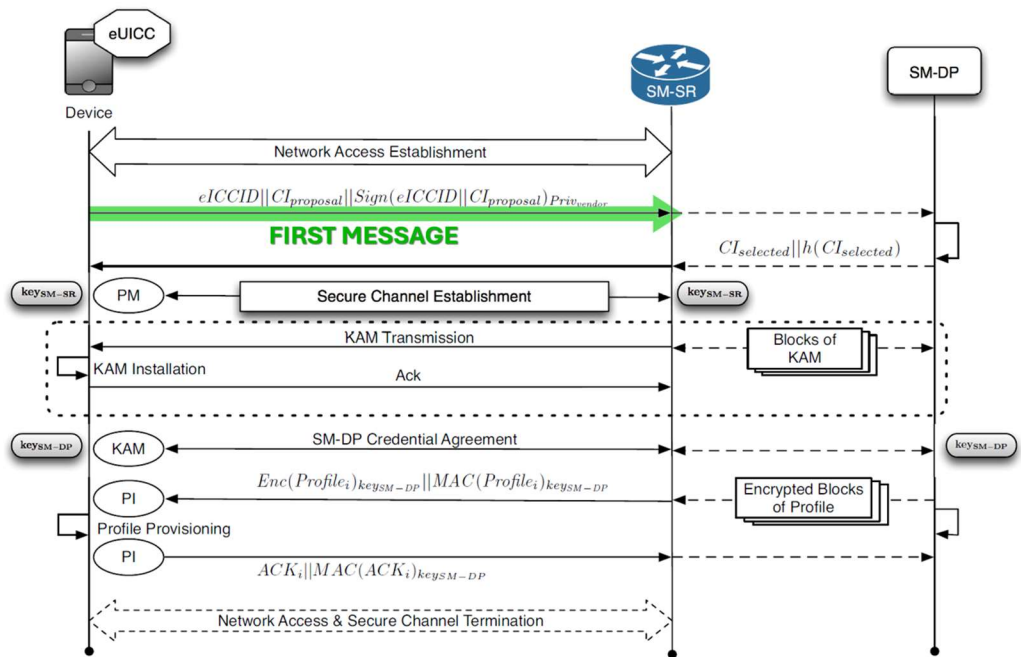
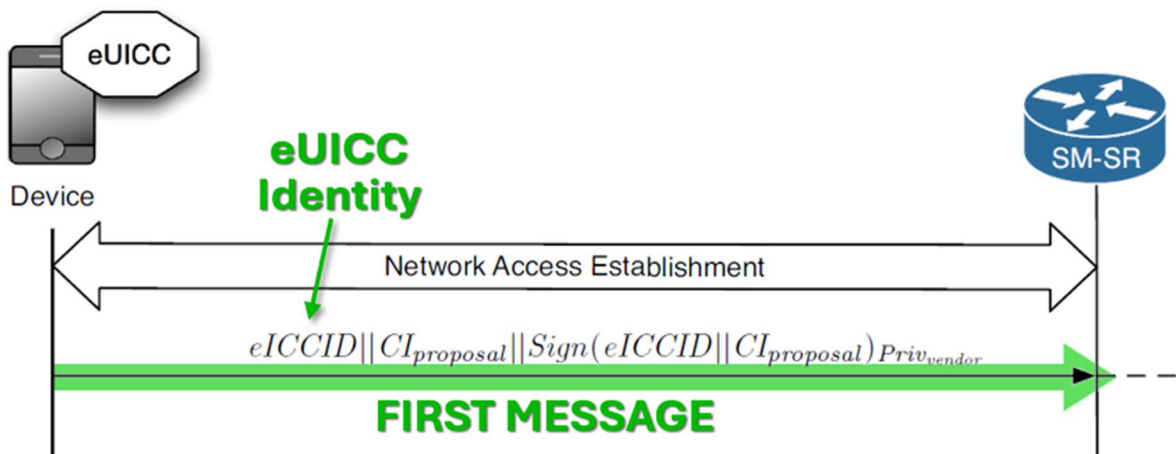


Fig. 4. Secure Profile Provisioning Procedure

c. Element 1[a][2][i]: (1) an identity of an embedded universal integrated circuit card (eUICC);

Park-GlobalPlatform-AbiChar teaches Element 1[a][2][i]. Ex-1002 ¶¶128-29.

As illustrated in Figure 4, excerpted below, Park’s capability negotiation message (the claimed “first message”) includes an “eUICC ID ($eICCID$).” Ex-1005, 300.



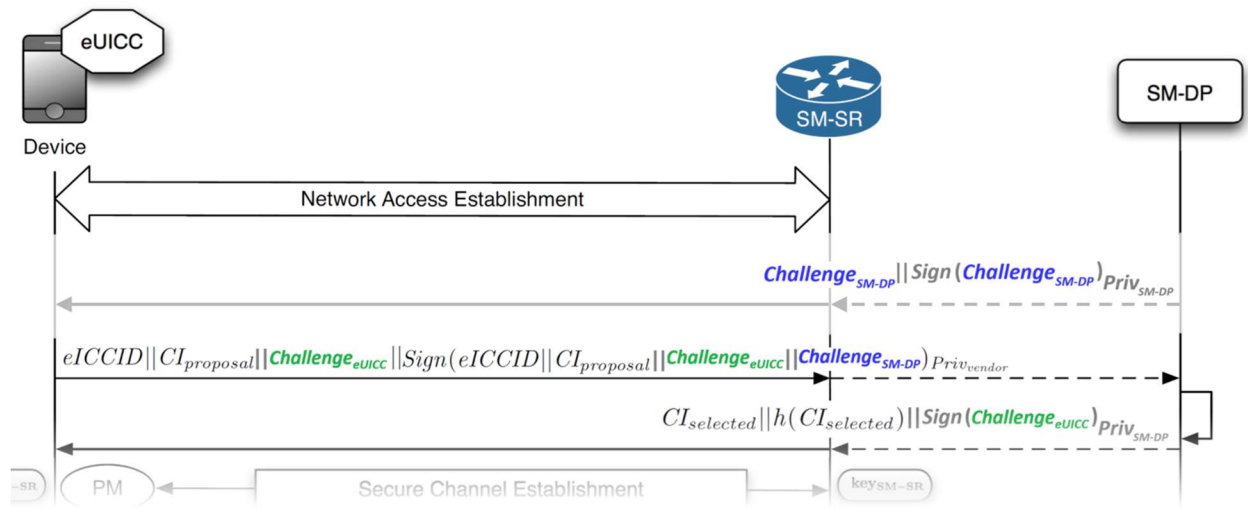
Ex-1005, Fig. 4 (excerpted)

Park explains that the “eUICC ID” is an “[i]dentification for eUICC to identify a particular eUICC.” *Id.*, 299. Thus, the eUICC ID within Park’s capability negotiation message (claimed “first message”) corresponds to the claimed “identity of an embedded universal integrated circuit card (eUICC).”

d. Element 1[a][2][ii]: (2) a nonce; and

Park-GlobalPlatform-AbiChar teaches Element 1[a][2][ii]. Ex-1002 ¶¶130-32.

As explained in Section XII.A.1, Park-GlobalPlatform-AbiChar would use GlobalPlatform’s PKI authentication scheme for authentication. This scheme involves issuing random challenges and verifying private key signatures applied to those challenges (Ex-1006, 246-47, 266-68, 271), and would have been applied to Park’s capability negotiation procedure, as illustrated in the modified version of Park’s Figure 4, below:

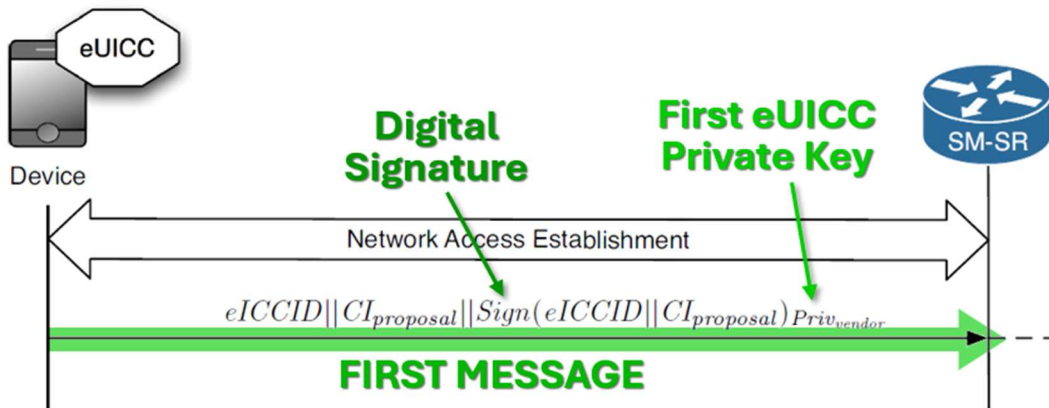


Ex-1002 ¶¶57-59, 131. Thus the SM-DP first issues a random challenge ($Challenge_{SM-DP}$, blue) to the eUICC. Then, the eUICC responds, by signing the SM-DP's challenge with its private key ($Sign(...Challenge_{SM-DP})Priv_{vendor}$) and issuing its own random challenge to the SM-DP ($Challenge_{eUICC}$, green)—both within the initial capability negotiation message, for efficiency purposes. Ex-1002 ¶132. A POSITA would have understood that each random challenge is a random number intended to be used only once—i.e., a nonce. Ex-1002 ¶¶66, 132. Thus, the eUICC's capability negotiation message (the claimed “first message”) includes two different nonces, as illustrated below—(1) the random challenge ($Challenge_{eUICC}$, green) issued to the SM-DP, and (2) the SM-DP's random challenge that has been signed with the eUICC's private key ($Sign(...Challenge_{SM-DP})Priv_{vendor}$) for subsequent use by the SM-DP in authenticating the eUICC. Ex-1002 ¶132.



- e. **Element 1[a][2][iii]: (3) a first digital signature, generated using a first eUICC private key, wherein the first eUICC private key corresponds to a first eUICC public key;**

Park-GlobalPlatform-AbiChar teaches Element 1[a][2][iii]. Ex-1002 ¶¶133-34. Park explains that, “[t]he eUICC ID and capability information are signed by the eUICC Vendors private key,” $Priv_{vendor}$ and this signed value, $Sign(\dots)_{Priv_{vendor}}$ is added to the message, as illustrated in excerpted Figure 4, below. Ex-1005, 300.

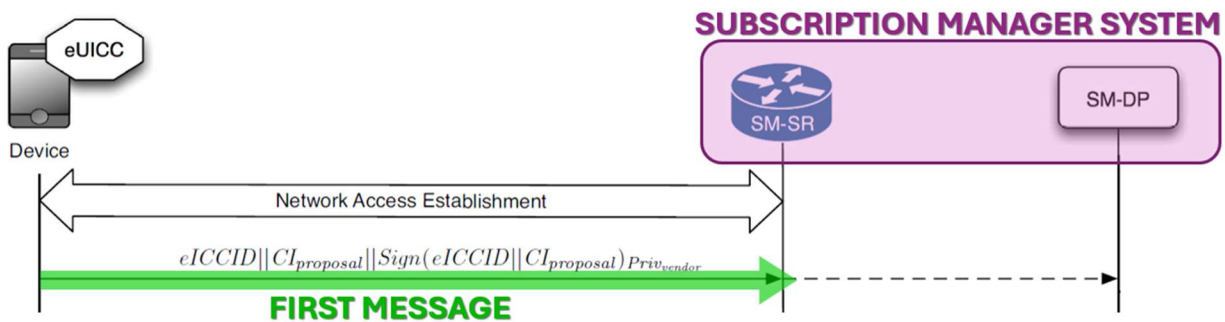


Park explains that the eUICC Vendor private key is part of a public key pair (i.e., it corresponds to an eUICC Vendor public key), and that it is generated by the eUICC Vendor and stored on the eUICC during the pre-provisioning stage. Ex-1005, 300.

- f. **Element 1[b]: (b) sending the first message via a network application to a subscription manager system;**

Park-GlobalPlatform-AbiChar teaches Element 1[b]. Ex-1002 ¶¶135-37.

Park teaches sending the capability negotiation message (claimed “first message”) from the eUICC to the SM-DP through the secure channel provided by the SM-SR, as illustrated in Figure 4, excerpted below. The SM-DP and SM-SR correspond to distinct logical and/or physical entities within Park’s Subscription Manager (SM).



As illustrated in Figure 5, below, messages sent from the eUICC to the SM are sent using a Device Module (DM), which is “software running in the Device to perform the roles of the intermediary between eUICC and SM-SR.” Ex-1005, 299. Park’s DM corresponds to the claimed “network application.”

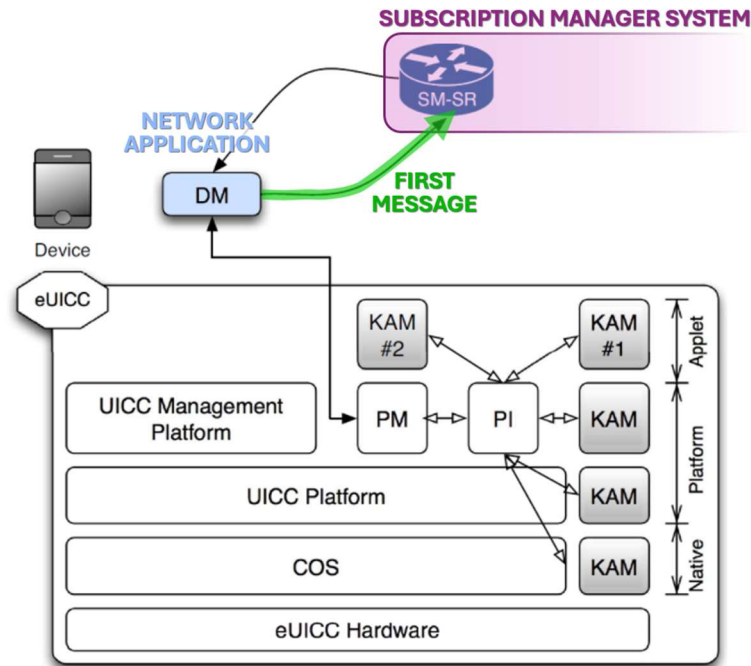
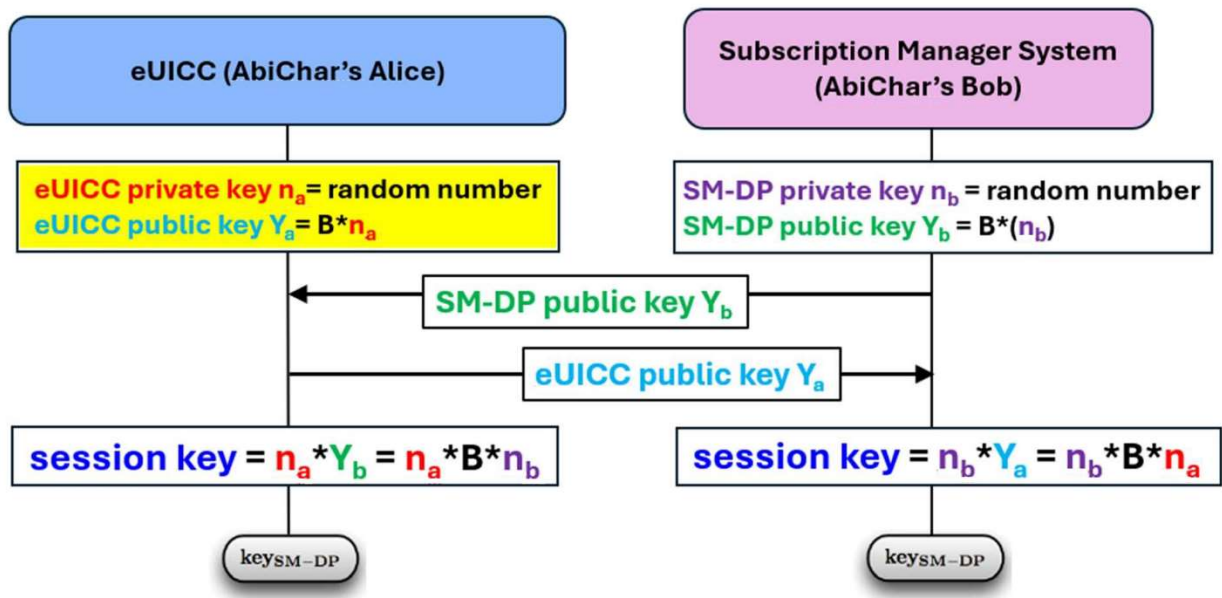


Fig. 5. Internal Architecture of eUICC

- g. **Element 1[c]: (c) deriving a **second eUICC private key** and a corresponding **second eUICC public key** using a first random number generator and a first set of cryptographic algorithms;**

Park-GlobalPlatform-AbiChar teaches Element 1[c]. Ex-1002 ¶¶138-140.

As explained in Section XII.A.1, the combined system uses AbiChar's ECDH key agreement protocol illustrated below:



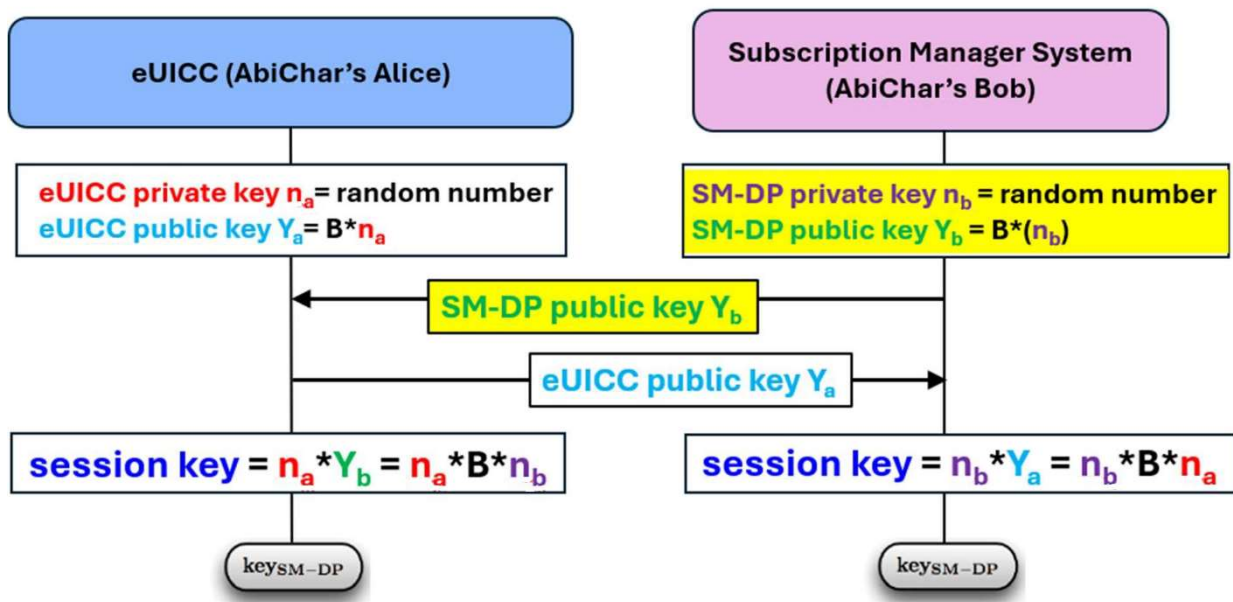
Ex-1002 ¶139.

The eUICC's **private key n_a** (claimed “second eUICC private key”) is generating using a random number. Ex-1007, 3; Ex-1002 ¶140. The eUICC calculates corresponding **public key $Y_a = B * n_a$** (claimed “second eUICC public key”) from **private key n_a** . Ex-1007, 3. The value B used to calculate the eUICC public key Y_a is a point on an elliptic curve. Ex-1007, 3. Thus, the eUICC public key (Y_a) is calculated from the randomly-generated eUICC private key (n_a), using elliptic-curve scalar multiplication—the fundamental cryptographic operation of ECDH. Ex-1002 ¶140.

- h. **Element 1[d]: (d) storing a **subscription manager public key** which corresponds to a **subscription manager private key**;**

Park-GlobalPlatform-AbiChar teaches Element 1[d]. Ex-1002 ¶¶141-43.

In AbiChar's ECDH key exchange, Park's SM-DP uses a random number to generate a **private key** n_b (claimed "subscription manager private key"), and then calculates a **corresponding public key** $Y_b = B * n_b$ (claimed "subscription manager public key") from that private key n_b . Ex-1007, 3.



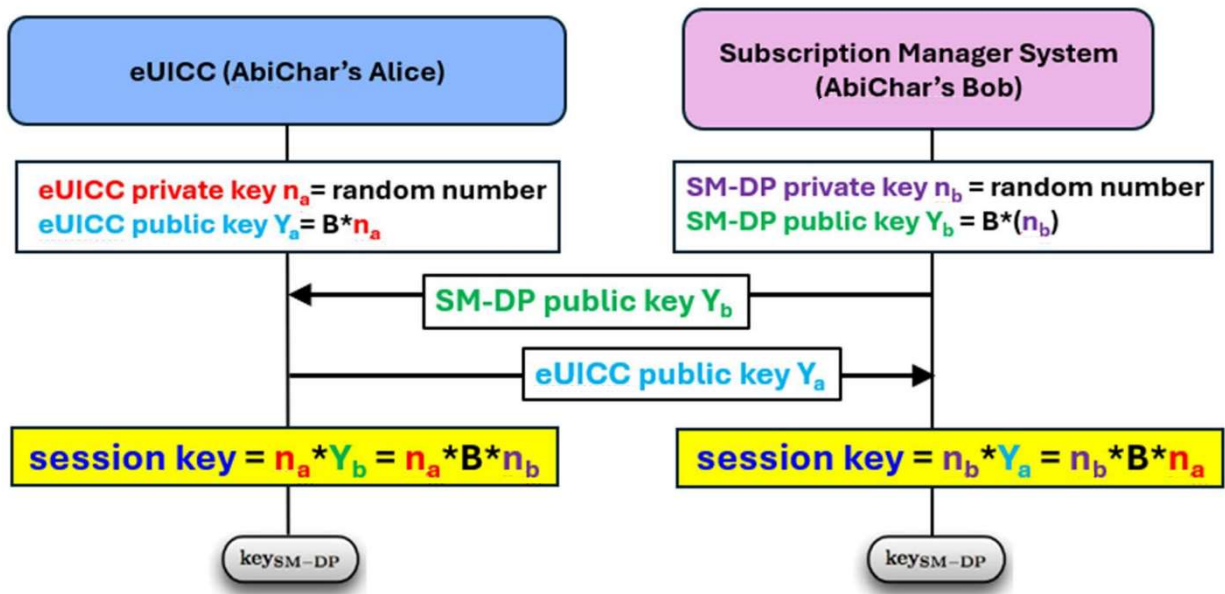
The SM-DP then sends its **public key** Y_b (claimed "subscription manager public key") for use by the eUICC in calculating the shared **session key** (SM-DP Credential). Ex-1007, 3; Ex-1005, 301. A POSITA would have understood that the eUICC would store the **public key** Y_b (claimed "subscription manager public key") at least long enough for the eUICC to use it in the key exchange algorithm. Ex-1002 ¶¶143.

i. Element 1[e][1]: (e) deriving a **profile key** using a key exchange algorithm based on at least:

(i) the **second eUICC private key**, and

(ii) the **stored subscription manager public key**,

Park-GlobalPlatform-AbiChar teaches Element 1[e][1]. Ex-1002 ¶¶144-46.



As illustrated above, the eUICC calculates the **shared SM-DP Credential/session key** (claimed “profile key”) using its **private key n_a** (claimed “second eUICC private key”), together with the SM-DP **public key Y_b** (claimed “subscription manager public key”) that it received from the SM-DP. Ex-1007, 3.

Similarly, the SM-DP calculates the **shared session key** using its **private key n_b** (claimed “subscription manager private key”), together with the eUICC **public key Y_a** (claimed “second eUICC public key”) that it received from the eUICC. Ex-1007, 3.

- j. **Element 1[e][2]: wherein the profile key can also be derived at the subscription manager system based at least on: the second eUICC public key, and the subscription manager private key;**

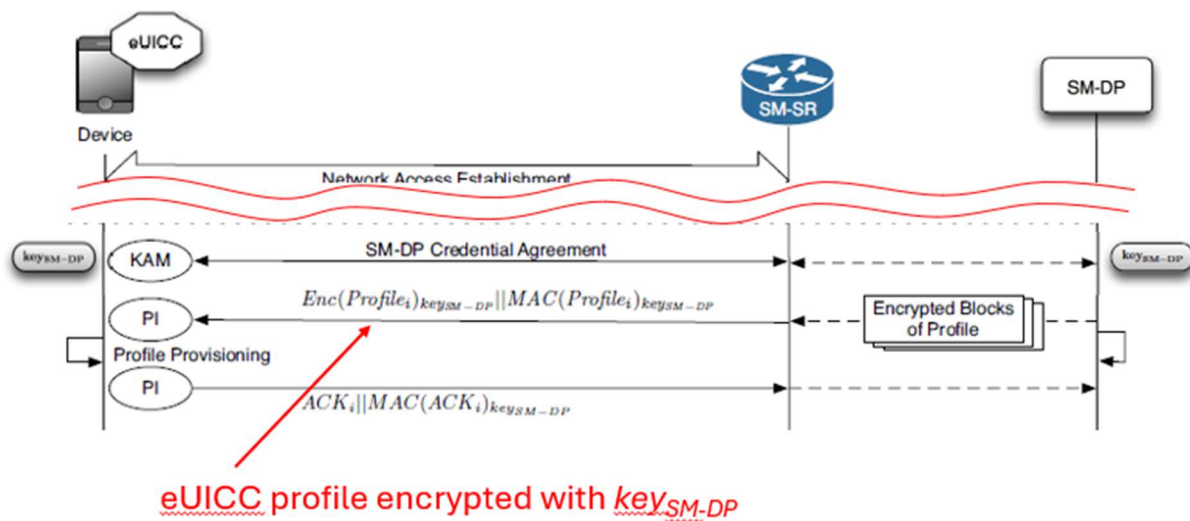
Similarly, the SM-DP calculates the shared session key using its private key n_b (claimed “subscription manager private key”), together with the eUICC public key Y_a (claimed “second eUICC public key”) that it received from the eUICC. Ex-1007, 3.

Thus, Park-GlobalPlatform-AbiChar teaches Element 1[e][2]. Ex-1002 ¶¶147-48.

- k. **Element 1[f]: receiving from the subscription manager system, an encrypted profile comprising a ciphertext including a key K encrypted with a symmetric key;**

Park-GlobalPlatform-AbiChar teaches Element 1[f]. Ex-1002 ¶¶149-152.

Park’s SM-DP, encrypts the profile “[u]sing the agreed SM-DP Credentials and the selected cryptographic algorithms,” $Enc(Profile_i)_{keySM-DP}$, and then “sends the protected profile blocks to the eUICC.” Ex-1005, 300.



Park teaches that profiles contain “security-sensitive information,” including “keys” used for accessing a mobile network. Ex-1005, 297. GlobalPlatform teaches that, when transmitting messages that include sensitive data, such as “secret or private keys,” an additional layer of protection is provided “over and beyond the security level required for the Secure Channel,” by first encrypting the sensitive data (using DEK), and then encrypting the entire message (including the encrypted sensitive data) using the session key, as illustrated below, using GlobalPlatform’s Figure F-11. Ex-1006, 258, 260, 264.

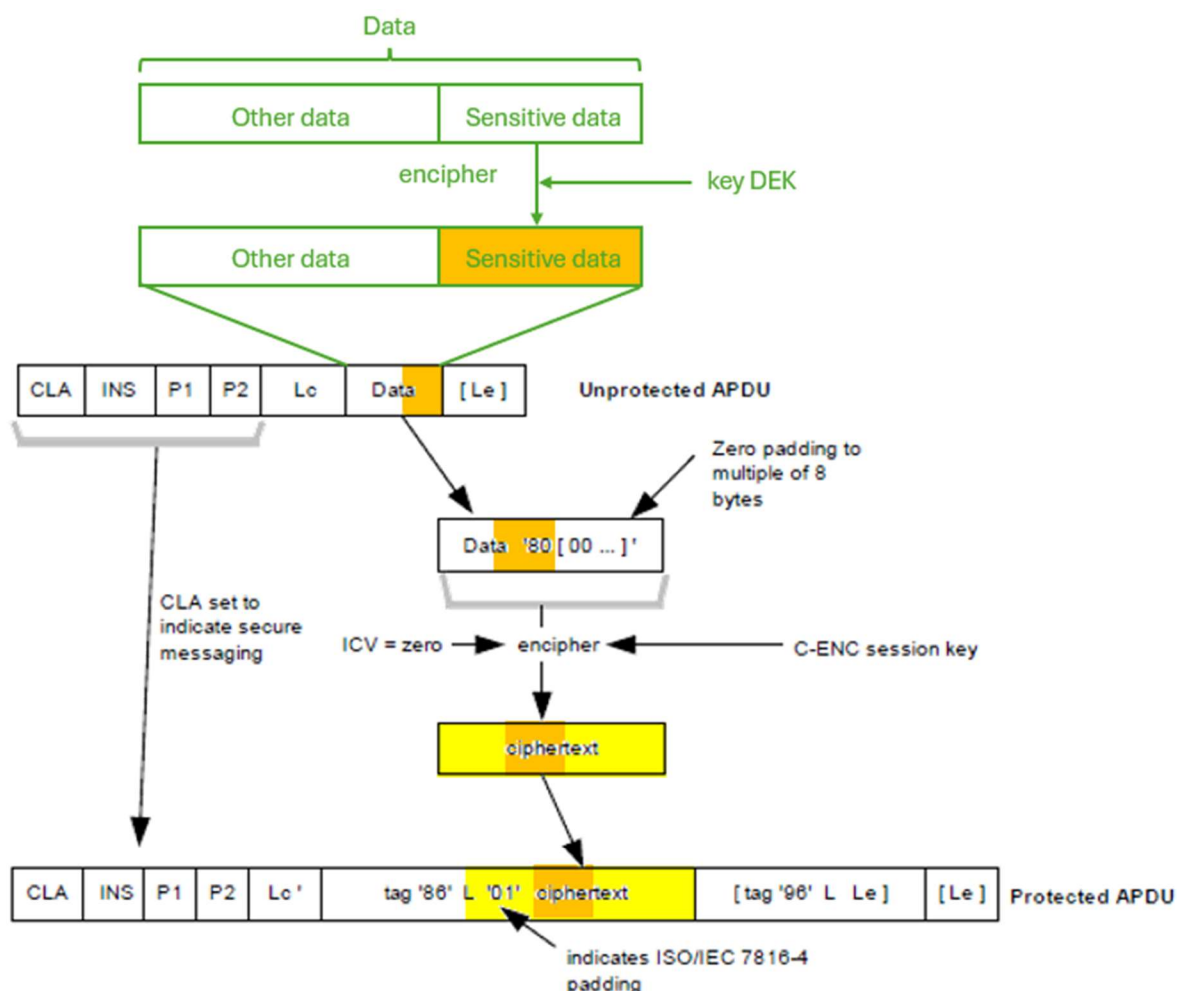


Figure F-11: Secure Messaging: Command message protected for confidentiality

GlobalPlatform’s DEK-based inner encryption applied to the sensitive keys included within Park’s profile adds an additional layer of protection for the most sensitive data within the profile. Ex-1002 ¶¶40-41, 150-51.

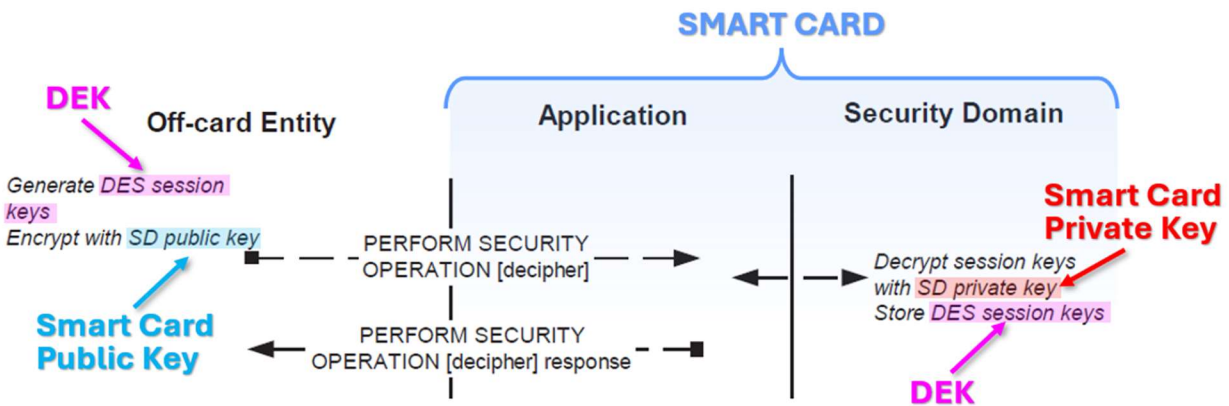
GlobalPlatform explains that encryption with the DEK “uses DES” (Ex-1006, 264), and that “[t]he Data Encryption Standard (DES) is a symmetric cryptographic algorithm that requires the use of the same secret key to encrypt and decrypt data.”

Id., 174. Thus, in the combined system, GlobalPlatform’s ***symmetric DEK*** would be used to encrypt a key K included within Park’s profile. Ex-1002 ¶¶33-35, 152.

I. Element 1[g]: (g) receiving the symmetric key;

Park-GlobalPlatform-AbiChar teaches Element 1[g]. Ex-1002 ¶¶153-55.

Because the DEK is a symmetric key, in order for the eUICC to decrypt a key K encrypted with the DEK, the eUICC must have its own copy of the DEK. Ex-1002 ¶¶33-35, 154. GlobalPlatform teaches a “key transport option” that may be used by an off-card entity to send the DEK to a smart card. Ex-1006, 251-52. The DEK is sent to the card using the “PERFORM SECURITY OPERATION [decipher] command.” Ex-1006, 255-56 (explaining that the PERFORM SECURITY OPERATION [decipher] command includes “control reference templates” that “contain the actual key values,” with “control reference template ‘B8’” used to transmit the DEK). GlobalPlatform further teaches that keys sent to smart card using the key transport option are encrypted using the card’s public key. Ex-1006, 247, 275-76. This is illustrated in Figure F-9, excerpted below:



Ex-1006, Fig. F-9 (excerpted)

A POSITA would have recognized that in the combined Park-GlobalPlatform system, the SM-DP would send the DEK (claimed “symmetric key”) to the eUICC, encrypted with the eUICC’s public key. Further, the SM-DP is part of the network trust zone separate from the SM-SR. Ex-1002 ¶¶40-41, 155.

- m. **Element 1[h]: (h) decrypting, by the embedded universal integrated circuit card, at least a portion of the encrypted profile using the profile key;**

Park-GlobalPlatform-AbiChar teaches Element 1[h]. Ex-1002 ¶¶156-57.

Park teaches that, upon receiving the encrypted profile block from the SM-DP, “PM of eUICC performs the necessary cryptographic operations for the secure communication channel, and then PI of eUICC decrypts and verifies the received profile blocks.” Ex-1005, 300, Table1 (“Profile Installer (PI) | The software running in the eUICC to perform the decryption and verification of the protected profile blocks from SM-DP”). Because each of the SM-DP and eUICC share the same SM-

DP Credential key_{SM-DP} (claimed “profile key”)⁵, a POSITA would have understood that the SM-DP Credential is a symmetric key—i.e., one that can be used for both encryption and decryption. Ex-1002 ¶¶33-35, 157. Thus, a POSITA would understand that decrypting each profile block, as performed by the eUICC, would involve using the **SM-DP Credential** key_{SM-DP} (claimed “profile key”). *Id.*

- n. Element 1[i]: (i) decrypting, by the embedded universal integrated circuit card, at least a portion of the ciphertext using the symmetric key; and**

Park-GlobalPlatform-AbiChar teaches Element 1[c][9]. Ex-1002 ¶¶158-59.

As described in Section XII.A.1, in the combined Park-GlobalPlatform system, certain key materials within the profile (e.g., authentication keys K) are encrypted under a DEK (the claimed “symmetric key”) before the entire profile is wrapped under the SM-DP Credential/session key. GlobalPlatform teaches that, in response to receiving such encrypted “sensitive data,” the smart card will perform a “decryption” process that “is the exact opposite” of the encryption process. Ex-1006, 264. Thus, once the eUICC has decrypted the outer profile layer using the SM-DP Credential/session key, it would use the received DEK to decrypt the inner ciphertext containing the key K. Ex-1002 ¶159.

⁵ Park’s SM-DP credential, or key_{SM-DP} , is the equivalent of GlobalPlatform’s C-ENC session key, which it uses to protect the profile data. Ex-1006, 258-64, Fig. F-11.

- o. Element 1[j]: (j) storing at least the key K in the embedded universal integrated circuit card for use in future communications.**

Park-GlobalPlatform-AbiChar teaches Element 1[j]. Ex-1002 ¶¶160-62.

Park teaches that, following successful decryption of the profile blocks, “the profile blocks are provisioned inside the eUICC.” Ex-1005, 300. A POSITA would have understood that this provisioning process would involve storing the key K in the eUICC, so that it can be used by the eUICC to access the mobile network corresponding to the provisioned profile. Ex-1005, 297 (MNO profile includes “keys”); Ex-1002 ¶¶77, 161.

Thus, Park-GlobalPlatform-AbiChar renders obvious Claim 1.

3. Dependent Claims 2-17, 19-22

- a. Claim 2: The method of claim 1, wherein the subscription manager public key is downloaded via the network application.**

Park-GlobalPlatform-AbiChar teaches Claim 2. Ex-1002 ¶¶163-64.

As explained for Element 1[d], during the key exchange between the eUICC and the SM-DP, the SM-DP sends the eUICC its **public key Y_b** (claimed “subscription manager public key”) for use by the eUICC in calculating the shared session key (SM-DP Credential). Ex-1007, 3; Ex-1005, 301. Park’s Figure 5, below, shows that the SM-DP sends messages to the eUICC “via the secure communication channel of SM-SR,” and messages sent via the SM-SR to the eUICC are received

using the eUICC’s Device Module (DM)—i.e., the claimed “network application.”

Ex-1005, 299-300.

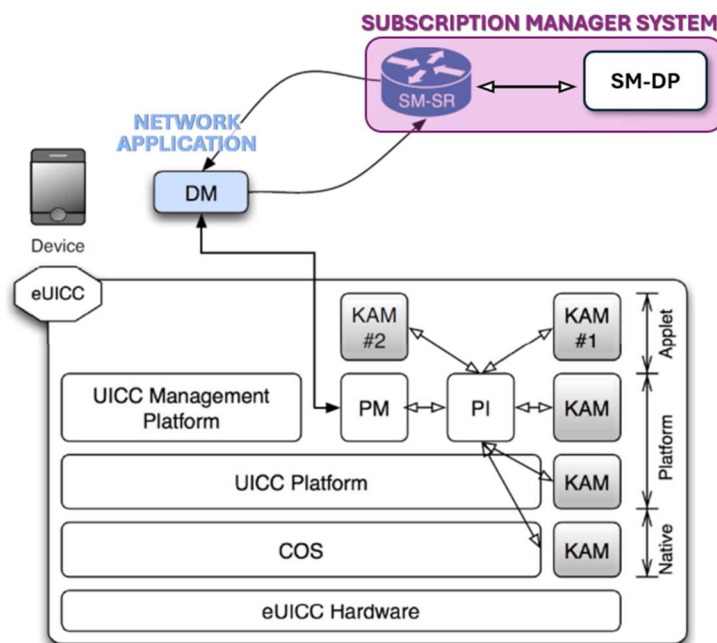


Fig. 5. Internal Architecture of eUICC

Ex-1005, Fig. 5 (modified to illustrated the interworking between the SM-SR and SM-DR).

- b. Claim 3: The method of claim 1, wherein the step of storing the **subscription manager public key** occurs at the time of manufacture of the embedded universal integrated circuit card.**

Park-GlobalPlatform-AbiChar teaches Claim 3. Ex-1002 ¶¶165-67.

AbiChar’s key-exchange protocol uses ephemeral key pairs. Ex-1006, 3. A POSITA, however, would have recognized that substituting a long-term static key pair for the server-side entity—corresponding here to Park’s SM-DP—could provide

practical advantages in the context of Park's secure profile-provisioning architecture. Ex-1002 ¶166. Park characterizes the eUICC as a "resource-constrain[ed]" device for which "overhead for the security-related operations" should be minimized. Ex-1005, 299. Thus, when the eUICC has limited computational and communication resources, it may be advantageous to simplify the key-exchange procedure by using a static key pair on the SM-DP side, thereby reducing the messaging required to establish the shared secret. Ex-1002 ¶166.

In such a modified ECDH scheme, the eUICC would use the **static SM-DP public key** to calculate the shared SM-DP Credential/session key prior to engaging in the key exchange, in the same manner as described in Section XII.A.2.i. Ex-1002 ¶167. A POSITA would have recognized that a natural way to provide the SM-DP public key to the eUICC would be "during the eUICC-based device manufacturing," as part of Park's "Pre-Provisioning Procedure." Ex-1005, 300; Ex-1002 ¶167. In particular, a POSITA would have recognized that it would be straightforward to add the SM-DP's private key to "the SM-SR Credentials (key_{SM-SR}) and the capability information ($CI_{installed}$) of each eUICC" that "are installed into each eUICC" during the pre-provisioning procedure. *Id.* Indeed, this would enable the eUICC would use the same SM-DP static public key to calculate the shared SM-DP Credential/session key as it would have used prior to engaging in the key exchange, to authenticate the

SM-DP (by verifying the SM-DP's private key signature on the challenge it issued to the SM-DP). *See* Element 1[a][2][iii].

- c. **Claim 4: The method of claim 1, wherein the key exchange algorithm is a Diffie-Hellman key exchange.**

Claim 5: The method of claim 1, wherein the key exchange algorithm is an Elliptic Curve Diffie-Hellman key exchange.

Park-GlobalPlatform-AbiChar teaches Claims 4-5. Ex-1002 ¶¶168-69.

AbiChar teaches a “Diffie-Hellman key exchange based on EC”—i.e., an Elliptic Curve Diffie-Hellman key exchange. Ex-1007, 3; Ex-1002 ¶169.

- d. **Claim 6: The method of claim 1, wherein the encrypted profile is encrypted in a manner so that intermediate nodes on an IP network would not be able to read data within the profile in an unencrypted form.**

Park-GlobalPlatform-AbiChar teaches Claim 6. Ex-1002 ¶¶170-71.

Park's secure profile provisioning architecture (SPA) provides end-to-end security for a profile sent from the SM-DP to the eUICC. Ex-1005, 297, 301. As Park recognized, because the SM-SR may be operated by an entity distinct from the SM-DP / MNO—e.g., a third-party—it therefore should not be trusted with access to sensitive profile contents. *Id.*, 301 (explaining that, for security purposes, “the SM-DP Credentials should not be revealed to any party, even SM-SR, except for eUICC”). Park's SPA addresses precisely this architectural challenge, defining an overall framework in which confidentiality between the SM-DP and the eUICC is

ensured by having the eUICC and SM-DP agree on profile encryption keys directly, without exposing those keys to the SM-SR or any other intermediaries. *Id.*, 299-301. This ensures that no intermediate nodes in the network (including the SM-SR) are able to read data within the profile. *Id.*; Ex-1002 ¶171.

- e. **Claim 7: The method of claim 1, wherein subscription manager system stores the ciphertext and the symmetric key before step (a).**

Park-GlobalPlatform-AbiChar teaches Claim 7. Ex-1002 ¶¶172-76.

- (1) the subscription manager system stores the ciphertext and the symmetric key**

Park's SM-DP prepares and delivers profiles enabling an eUICC to access an MNO's mobile network. Ex-1005, 297. A POSITA would have recognized that for security reasons, the MNO-specific network-access credentials, such as key K included within the profiles would not be transmitted or stored in plaintext form when exchanged between the MNO and the SM-DP, or otherwise handled during profile preparation. Ex-1002 ¶173 (citing Ex-1006 §F.3.2.7; Ex-1024, 2). Rather, this sensitive data (including key K) would be protected in ciphertext form—that is, encrypted under a symmetric data-encryption or key-wrapping key (such as DEK))—to ensure the secure transfer of the key material. Ex-1002 ¶173.

A POSITA would further have recognized that, once the SM-DP had obtained the encrypted key K, there would be no need for the SM-DP to decrypt and then re-encrypt it. Ex-1002 ¶174. Instead, to maintain end-to-end key protection and

minimize exposure of sensitive key material, the SM-DP would simply retain the encryption on key K and include that encrypted key within the profile data it delivered to the eUICC, securely transported under the SM-DP Credential/session key. Ex-1002 ¶174.

A POSITA would also have understood that the symmetric key-wrapping key (DEK) used to encrypt key K must ultimately be available to the eUICC in order for the eUICC to decrypt the ciphertext and recover key K. Ex-1002 ¶175. A POSITA would have recognized that the MNO could transmit the DEK either directly to the eUICC, or via the SM-DP, as part of the profile-provisioning process. *Id.* A POSITA would have understood that having the SM-DP send the DEK to the eUICC would be appropriate in situations where the MNO has delegated control over the timing of profile installation to the SM-DP, because relying on a single entity (the SM-DP) to provision all profile-related data is operationally more efficient than requiring the eUICC to receive data from two separate entities (the SM-DP and the MNO). *Id.* In such implementations, the SM-DP would store the DEK together with the corresponding profile (including key K in ciphertext form) so that both could be securely provisioned to the eUICC during installation. *Id.*

(2) the subscription manager's storage of the ciphertext and the symmetric key occurs before step a)

A POSITA would further have recognized that the SM-DP would logically prepare and store the profile (together with the symmetric key (DEK) needed to decrypt the ciphertext), before the provisioning process is initiated. Ex-1002 ¶176. This sequencing would allow the SM-DP to prepare and stage the complete profile package in advance, enabling the provisioning session with the eUICC to proceed efficiently once initiated. *Id.*

- f. **Claim 8: The method of claim 1, wherein after the step (b), the embedded universal integrated circuit card is authenticated by the subscription manager system using at least the identity of the embedded universal integrated circuit card included in the first message.**

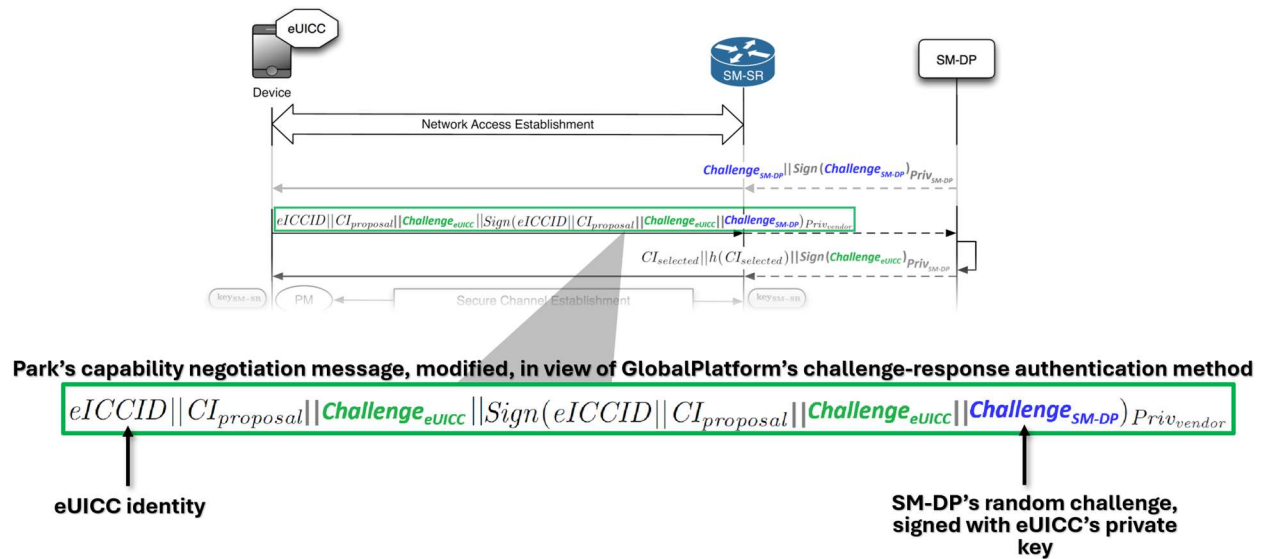
Claim 9: The method of claim 1, wherein after the step (b), the embedded universal integrated circuit card is authenticated by the subscription manager system using at least the digital signature included in the first message.

Park-GlobalPlatform-AbiChar teaches Claims 8-9. Ex-1002 ¶¶177-181.

Park teaches that, in response to receiving the capability negotiation message from the eUICC, the SM “acquires the eUICC Vendors certificate from the eUICC Certification Center,” and, using that certificate, the “SM-DP verifies the signature of the eUICC ID and the capability information.” Ex-1005, 300. A POSITA would

have understood that the SM would use the eUICC ID included within the first message to identify the corresponding eUICC Vendors certificate. Ex-1002 ¶178.

As explained in Section XII.A.1, in the combined Park-GlobalPlatform-AbiChar system, Park's capability negotiation process is modified to include GlobalPlatform's challenge-response authentication method, as illustrated below:



In the combined system, the capability negotiation message sent from the eUICC (claimed “first message”) includes the eUICC’s response to a random challenge ($Challenge_{SM-DP}$) issued by the SM-DP—namely, the SM-DP’s challenge signed/encrypted with the eUICC’s private key ($Sign(...Challenge_{SM-DP})Priv_{eUICC}$) (claimed “digital signature”). Ex-1006, 246-47, 266-68, 271; Ex-1002 ¶180. To authenticate the eUICC, the SM-DP uses the eUICC’s public key—retrieved from the eUICC Vendors certificate—to decrypt the signed challenge. If the decrypted challenge matches the challenge issued to the eUICC, this verifies that the eUICC is

the authentic holder of the eUICC private key, thereby authenticating the eUICC to the SM-DP. Ex-1002 ¶180.

Thus, in the combined system, the SM-DP would use both the eUICC ID (Claim 8) and the eUICC's digital signature (Claim 9) from the capability negotiation message received from the eUICC to authenticate the eUICC. Ex-1002 ¶181.

g. Claim 10: The method of claim 1, wherein the decrypted ciphertext is stored in nonvolatile memory of the embedded universal integrated circuit card.

Park-GlobalPlatform-AbiChar teaches Claim 10. Ex-1002 ¶¶182-83.

As explained for Element 1[j], Park teaches that, following successful decryption of the profile blocks—which would include decryption of the ciphertext containing key K encrypted with the symmetric key—“the profile blocks are provisioned inside the eUICC.” Ex-1005, 300. A POSITA would have understood that this provisioning process would involve storing the decrypted profile data, including key K, within the eUICC's non-volatile (i.e., persistent) memory so that it remains available for use in accessing the associated mobile network after power loss or device reset. Ex-1002 ¶183.

h. Claim 11: The method of claim 1, wherein the embedded universal integrated circuit card is operably connected to a module and wherein the module is a machine-to-machine module.

Park-GlobalPlatform-AbiChar teaches Claim 11. Ex-1002 ¶¶184-85.

Park teaches that an initial use for eUICCs was “for the M2M (Machine-to-Machine) device.” Ex-1005, 297.

- i. **Claim 12: The method of claim 11, wherein the module comprises a module of a type selected from the group consisting of: vending machine, automobile, alarm system, tracking device, monitoring device, security system and remote control device.**

Park-GlobalPlatform-AbiChar teaches Claim 12. Ex-1002 ¶¶186-87.

Park teaches that eUICCs have been “adopted into the small device such as smart meter, etc.” Ex-1005, 297. A POSITA would have understood that a smart meter is a type of monitoring device. Ex-1002 ¶187.

- j. **Claim 13: The method of claim 1, wherein the embedded universal integrated circuit card is operably connected to a module and wherein the module is a wireless module.**

Claim 14: The method of claim 13, wherein the module is configured to communicate using one or more wireless technologies of types selected from the group consisting of: WiFi, WiMax, General Packet Radio Services, Enhanced Data rates for GSM Evolution and 3rd Generation Partnership Project technology.

Park-GlobalPlatform-AbiChar teaches Claims 13-14. Ex-1002 ¶¶188-190.

Park teaches that the eUICC communicates with the SM using a “Device Module (DM),” as illustrated in annotated Figure 5, below. Ex-1005, 299.

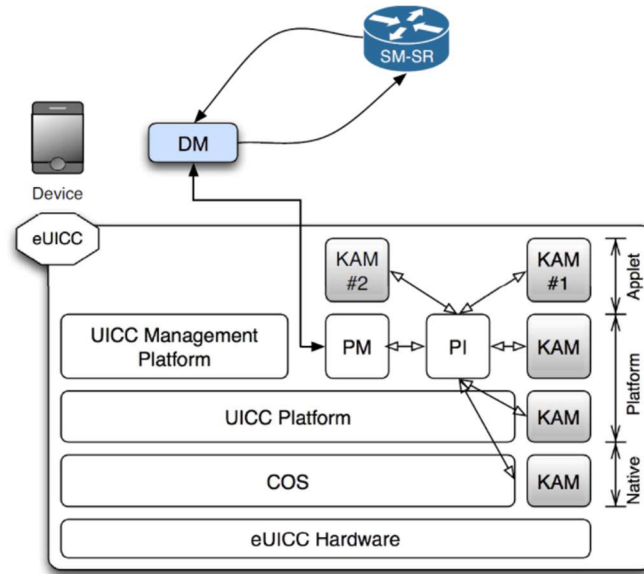


Fig. 5. Internal Architecture of eUICC

Park further teaches that this communication may occur over a wireless network, such as “WiFi,” or a “mobile network” accessible to the eUICC through “the provisioning profile.” Ex-1005, 300.

- k. **Claim 15:** The method of claim 13, wherein the module comprises a module of a type selected from the group consisting of the following: wireless handset, cellular phone, smartphone, laptop computer, tracking device, computer with a radio, circuit board with a radio, hand-held device, netbook, portable computer, multiprocessor system, microprocessor based consumer electronic, programmable consumer electronic, network personal computer, minicomputer, mainframe computer and smartcard.

Park-GlobalPlatform-AbiChar teaches Claim 15. Ex-1002 ¶¶191-92.

Park teaches that the field of use of the eUICC has extended to include “Consumer Electronic Devices.” Ex-1005, 297. Indeed, as illustrated in Figure 5, below, Park illustrates its eUICC as being housed within a smartphone:

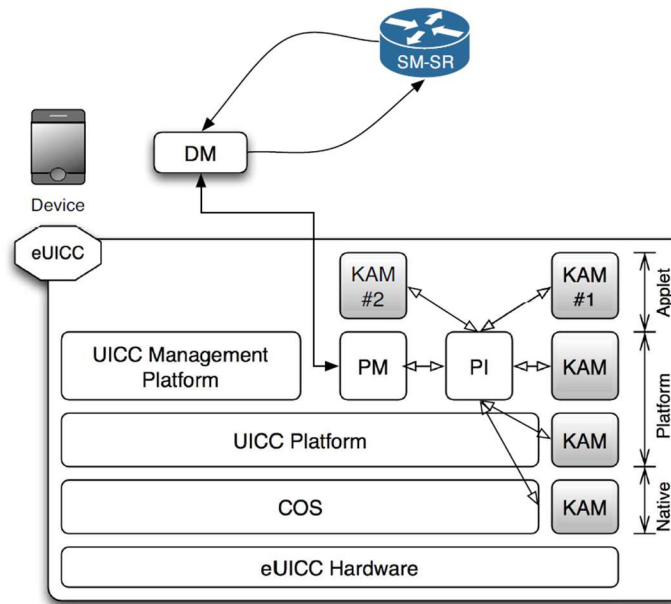


Fig. 5. Internal Architecture of eUICC

- I. **Claim 16:** The method of claim 1, wherein the embedded universal integrated circuit card is operably connected to a module, wherein the module is a wired module.

Claim 17: The method of claim 16, wherein the module is configured to communicate using one or more wired connections of types selected from the group consisting of: Ethernet, fiber optic connection and Universal Serial Bus connection.

Park-GlobalPlatform-AbiChar teaches Claims 16-17. Ex-1002 ¶¶193-94.

Park teaches that the eUICC may communicate with the SM-SR using “USB[], without a Provisioning Profile.” Ex-1005, 300. A POSITA would have recognized that a USB interface is a wired interface. Ex-1002 ¶194.

m. Claim 19: The method of claim 1, wherein the deriving step (c) is performed prior to the generating step (a).

Claim 20: The method of claim 1, wherein the generating step (a) is performed before the deriving step (c).

Park-GlobalPlatform-AbiChar teaches Claims 19-20. Ex-1002 ¶¶195-97.

A POSITA would have understood that the timing of when the eUICC generates its ephemeral key pair for use in the key exchange—whether before or after sending the initial capability negotiation message—would have been a simple matter of design choice. Ex-1002 ¶196.

A POSITA would have recognized that a designer could implement the ephemeral key generation either (1) close in time to when those keys are needed by the key exchange algorithm, or (2) earlier in the process (e.g., before sending the capability negotiation message), and that the choice between the two would involve a tradeoff between security and performance. Ex-1002 ¶197. For instance, generating the keys immediately before the key exchange increases security by limiting the private key’s lifetime and thereby reducing its exposure, but can introduce delay into the provisioning process. On the other hand, generating the keys earlier avoids such delays, but extends the private key’s lifetime and therefore

increases security risk. *Id.* A POSITA would have appreciated that system designers could flexibly adopt either approach, depending on whether minimizing delay or maximizing key protection was the higher operational priority. *Id.*

- n. **Claim 21: The method of claim 1, wherein the embedded integrated circuit card is operably connected to a module and a user accesses a user interface of the module to enter authentication information before the module receives the symmetric key.**

Park-GlobalPlatform-AbiChar teaches Claim 21. Ex-1002 ¶¶198-200.

GlobalPlatform teaches that a smart card or secure element (such as an eUICC) may be protected by a “Global PIN” that provides “a mechanism for a Cardholder Verification Method,” to prevent “access violations.” Ex-1006, 71. A POSITA would have understood that this operation involves the eUICC being operably connected to a host module—such as a handset, terminal, or other interface device—that provides a user interface (e.g., keypad or touchscreen) through which the user can enter the Global PIN. Ex-1002 ¶199.

A POSITA would have understood that requiring a user to enter the Global PIN before provisioning the eUICC with secret keys, such as the DEK (claimed “symmetric key”), would provide an additional layer of protection by ensuring that only an authorized user or operator could trigger a “PERFORM SECURITY

OPERATION [decipher] command” (Ex-1006, 255-56) to load a new DEK onto the eUICC. Ex-1002 ¶200.

- o. Claim 22: The method of claim 1, wherein the subscription manager receives the ciphertext from a mobile network operator before the network application sends the first message.**

Park-GlobalPlatform-AbiChar teaches Claim 22 for the same reasons as explained for Claim 7. Ex-1002 ¶201.

B. Ground 2: Claims 1-17 and 19-22 are obvious over Park in view of GlobalPlatform and X9.63-Overview.

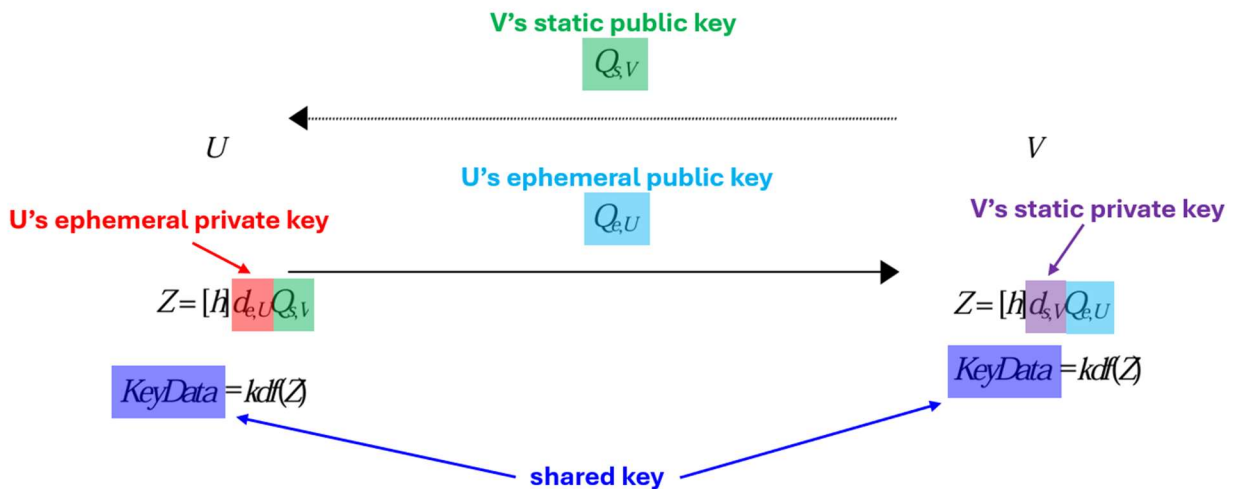
- 1. A POSITA would have been motivated to combine Park’s teachings with GlobalPlatform’s and X9.63-Overview’s teachings and would have had a reasonable expectation of success.**

A POSITA would have been motivated to combine Park’s secure profile provisioning architecture (SPA) with GlobalPlatform’s procedures for mutual authentication and sensitive data encryption, for the same reasons as explained for Ground 1, above. A POSITA would further have been motivated to combine Park’s SPA with X9.63-Overview’s 1-pass Diffie-Hellman key agreement protocol, to implement a specific embodiment of Park’s asymmetric key-exchange mechanism that is well-suited for Park’s architectural environment, in which eUICC’s are considered “resource-constrain[ed]” devices, for which “overhead for the security-related operations” is undesirable.” Ex-1005, 299. Ex-1002 ¶202.

As X9.63-Overview illustrates, and as described above in Section XI.D, the parties exchange public keys, $Q_{s,v}$ and $Q_{e,u}$, and then each side combines the other side's public key with its own private key—for example, the eUICC uses its **random private key** $d_{e,u}$ together with the **received public key** $Q_{s,v}$ —to generate a shared secret,

Z

ANSI X9.63 - 1-Pass DH



Ex-1008, 12. After deriving the shared secret Z, each party can derive the same symmetric key, KeyData, corresponding to the profile key, as described for 1(e). A POSITA would have understood that a one-pass DH exchange offers forward secrecy to the eUICC side of the communication. Ex-1002 ¶204. In the one-pass approach, the eUICC generates a fresh ephemeral public/private key pair for each

session, ensuring that even if its long-term private key were later compromised, previously provisioned profiles could not be decrypted. Ex-1002 ¶204.

2. Independent Claim 1

a. Elements 1[pre]-1[b] and 1[f]-1[j]

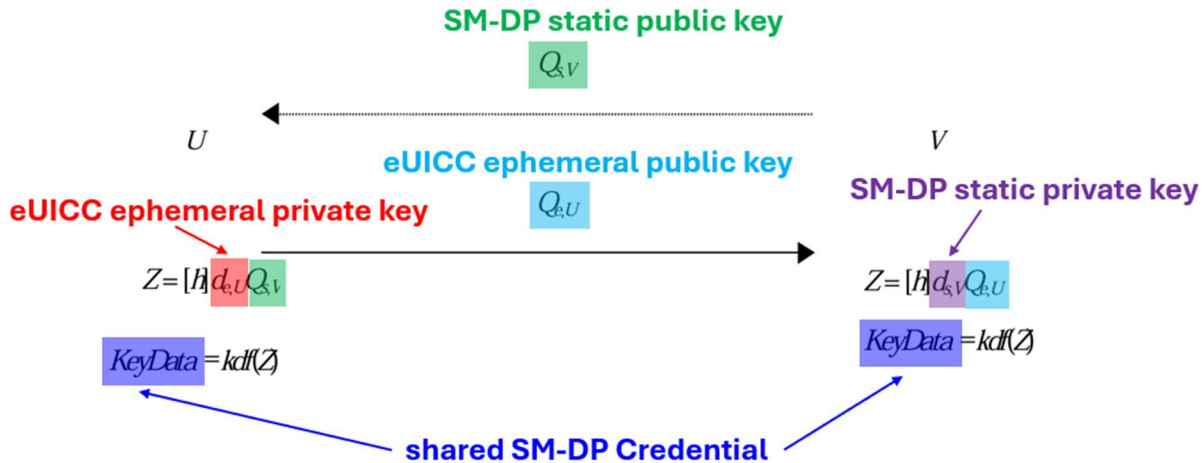
Park in view of GlobalPlatform and X9.63-Overview (“Park-GlobalPlatform-X9.63-Overview”) teaches Elements 1[pre] through 1[b], and 1[f] through 1[j], for the same reasons explained for Ground 1. Ex-1002 ¶205.

b. Element 1[c]: (c) deriving a **second eUICC private key** and a corresponding **second eUICC public key** using a first random number generator and a first set of cryptographic algorithms;

Park-GlobalPlatform-X9.63-Overview teaches Element 1[c]. Ex-1002 ¶¶206-08.

As explained in Section XII.B.1, a POSITA would have been motivated to use X9.63-Overview’s 1-pass Diffie-Hellman key exchange, illustrated below, to implement Park’s “asymmetric algorithm-based key agreement protocol,” as illustrated below:

ANSI X9.63 - 1-Pass DH



Ex-1008, 12. As illustrated, in the combined Park-GlobalPlatform-X9.63-Overview system, the eUICC generates an ephemeral private key $d_{e,U}$ (claimed “second eUICC private key”). *Id.* X9.63-Overview teaches that ephemeral private key $d_{e,U}$ is “selected randomly.” *Id.*, 7. A POSITA would have understood that randomly selecting a number involves using a random number generator. Ex-1002 ¶208. The eUICC then calculates a corresponding ephemeral public key $Q_{e,U}$ (claimed “second eUICC public key”). Ex-1007, 3. X9.63-Overview “[s]pecifies key agreement and key transport schemes using elliptic curve cryptography.” Ex-1008, 3. Accordingly, a POSITA would have understood that calculating public key $Q_{e,U}$ (claimed “second eUICC public key”) would involve applying the appropriate elliptic-curve cryptographic algorithm to the randomly selected private key $d_{e,U}$. Ex-1002 ¶208.

- c. **Element 1[d]: (d) storing a subscription manager public key which corresponds to a subscription manager private key;**

Park-GlobalPlatform-X9.63-Overview teaches Element 1[d]. Ex-1002 ¶¶209-210.

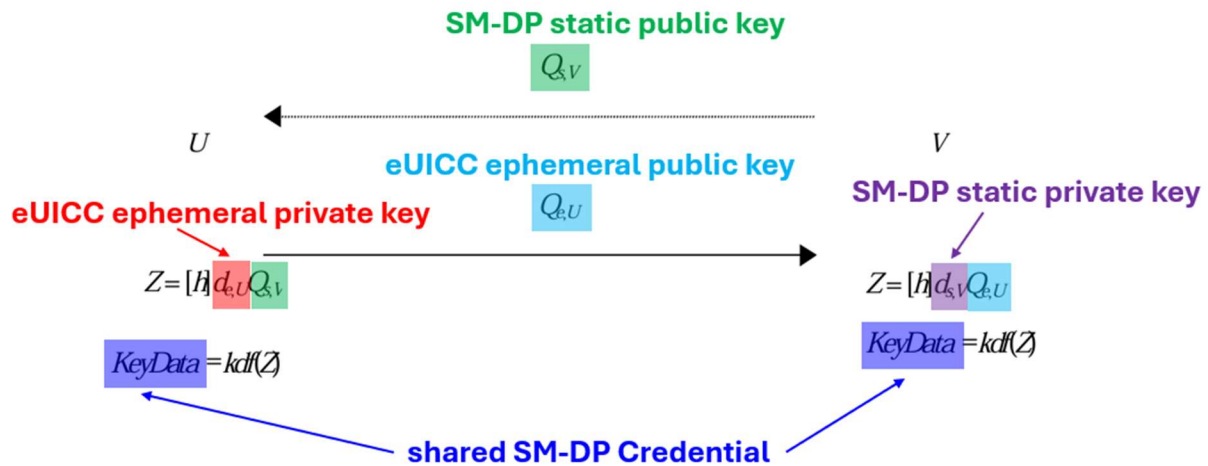
As explained in Section XII.B.1, in the combined system, X9.63-Overview's 1-pass Diffie-Hellman protocol employs SM-DP's static public-private key pair—i.e., static private key $d_{s,v}$ (claimed “subscription manager private key”), and static public key $Q_{s,v}$ (claimed “subscription manager public key”). Ex-1007, 12. This scheme assumes that the eUICC has been pre-provisioned with (i.e., stores) the **SM-DP's static public key**. Ex-1008, 12; Ex-1002 ¶210. Indeed, in the combined system, the eUICC would use the same **SM-DP static public key** to calculate the shared SM-DP Credential/session key as it would have used prior to engaging in the key exchange, to authenticate the SM-DP (by verifying the SM-DP's private key signature on the challenge it issued to the SM-DP). *See* Element 1[a][2][iii]; Ex-1002 ¶210.

- d. **Element 1[e][1]: (e) deriving a profile key using a key exchange algorithm based on at least:**
 - (i) the **second eUICC private key**, and
 - (ii) the **stored subscription manager public key**,

Park-GlobalPlatform-X9.63-Overview teaches Element 1[e][1]. Ex-1002 ¶¶211-13.

In the combined system, each party uses its private key and the other party's public key to generate a shared SM-DP Credential/session key (claimed "profile key") as illustrated in the diagram below:

ANSI X9.63 - 1-Pass DH



Ex-1008, 12. Thus, the eUICC calculates the **shared SM-DP Credential/session key** (claimed "profile key") using its private key $d_{e,U}$ (claimed "second eUICC private key"), together with the stored SM-DP public key $Q_{s,V}$ (claimed "subscription manager public key"). *Id.*

- e. **Element 1[e][2]:** wherein the **profile key** can also be derived at the subscription manager system based at least on: the **second eUICC public key**, and the **subscription manager private key**;

Similarly, the SM-DP calculates the **shared SM-DP Credential/session key** (claimed "profile key") using its private key $d_{s,V}$ (claimed "subscription manager

private key”), together with the eUICC public key $Q_{e,U}$ (claimed “second eUICC public key) that it received from the eUICC. Ex-1008, 12.

Thus, Park-GlobalPlatform-X9.63-Overview teaches Element 1[e][2]. Ex-1002 ¶¶214-15.

Accordingly, Park-GlobalPlatform-X9.63-Overview renders obvious Claim 1.

3. Dependent Claims 2-17, 19-22

a. Claims 2-3, 6-17, 19-22

Park-GlobalPlatform-X9.63-Overview teaches Claims 2-3, 6-15, 17-20 for the same reasons explained for Ground 1. Ex-1002 ¶217.

b. Claim 4: The method of claim 1, wherein the key exchange algorithm is a Diffie-Hellman key exchange.

Claim 5: The method of claim 1, wherein the key exchange algorithm is an Elliptic Curve Diffie-Hellman key exchange.

Park-GlobalPlatform-X9.63-Overview teaches Claims 4-5. Ex-1002 ¶¶218-19.

X9.63-Overview teaches a “1-Pass” Diffie-Hellman key exchange (Ex-1008, 12) that is included in ANSI X9.63’s set of “key agreement and key transport schemes using elliptic curve cryptography” (Ex-1008, 3). Thus, X9.63-Overview teaches a Diffie-Hellman key exchange that uses “elliptic curve cryptography”—i.e., an Elliptic Curve Diffie-Hellman key exchange. Ex-1008, 3, 12.

C. Grounds 3-4: Claim 18 is obvious over Park in view of GlobalPlatform, AbiChar, and Weiss, and Park in view of GlobalPlatform, X9.63-Overview, and Weiss

- 1. A POSITA would have been motivated to add Weiss's teachings to the Park-GlobalPlatform-AbiChar and Park-GlobalPlatform-X9.63-Overview systems and would have had a reasonable expectation of success.**

Weiss teaches that “secure elements, such as SIMs, eUICCs, UICCs” (Ex-1009, ¶3) may be included in a variety of devices, “such as a tablet or notebook computer, a TV system, a set top box, a vending machine, a vehicle, a surveillance camera, a sensor device and the like” (*id.*, ¶31). A POSITA seeking to broaden the applicability of Park's eUICC provisioning architecture beyond the specific implementations identified in Park would have looked to references such as Weiss, which expressly teaches that secure elements can be embedded in numerous types of connected or networked devices. Ex-1002 ¶220. Incorporating Park's eUICC into Weiss's disclosed devices would have amounted to a straightforward substitution of one known secure element implementation for another, yielding predictable results. Ex-1002 ¶220.

- 2. Claim 18: The method of claim 16, wherein the module comprises a module of a type selected from the group consisting of: computer, security camera, security monitoring device and networked controller.**

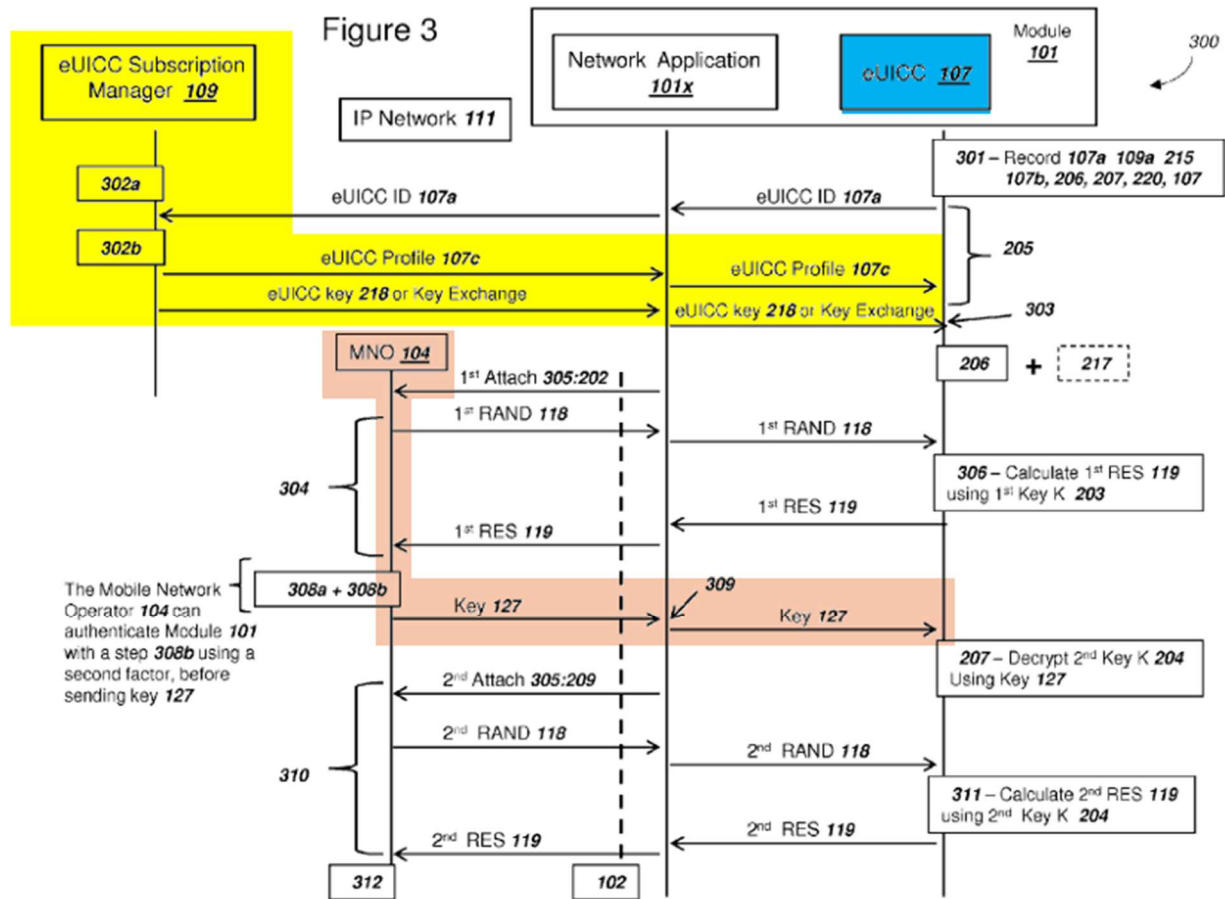
Park-GlobalPlatform-AbiChar-Weiss and Park-GlobalPlatform-X9.63-Overview-Weiss teach Claim 18. Ex-1002 ¶¶221-22.

Weiss teaches that an eUICC may be included in “a tablet or a notebook computer.” Ex-1009, ¶31.

D. Ground 5: Claims 1-22 are obvious over Nix175 in view of Park and GlobalPlatform

1. A POSITA would have been motivated to combine Nix175, Park, and GlobalPlatform and would have had a reasonable expectation of success.

Nix175 discloses the claims of the '094 Patent with the exception of claim element 1(g) “receiving the symmetric key,” to the extent PO alleges that receiving the symmetric key includes receiving it from the subscription manager (SM). As discussed above in Sections **Error! Reference source not found.** and **Error! Reference source not found.**, the '094 Patent (and Nix175) introduces the claimed “symmetric key” to address a security concern that arises if distribution and release of the profile (including decryption of the secret key K) were to be under the control of the SM and not the MNO. Ex-1016, 3:32-36, 34:27-45; Ex-1002 ¶223. Nix175’s solution to the security problem allows the SM to handle delivery the encrypted profile but requires that the MNO handle delivery of the symmetric key, without which the eUICC cannot decrypt the portion of the profile containing key K. *Id.* 34:54-60. This architecture is depicted in Figure 3 of Nix175, annotated below:



Above, the subscription manager (yellow) provides the encrypted eUICC profile 107c to the eUICC (blue). But the eUICC cannot decode secret key K (“2nd Key K 204”) until it receives symmetric key 127 (red) from the MNO 104. Ex-1016, Fig. 3; Ex-1002 ¶224.

While Nix175’s introduction of a symmetric key controlled and delivered by the MNO addresses the security issue by preventing the SM from being able to access the secure key K, a POSITA would have recognized that it imposes an undue burden on the MNO, which now must handle key management and delivery for

every eUICC. Ex-1002 ¶225. Thus a POSITA would have been motivated to look to Park, which teaches a split architecture for the subscription manager that divides it into two parts: “SM-DP (Data Preparation)” for generating and encrypting the operator profile, and “SM-SR (Secure Routing)” for transporting and installing the encrypted profile onto the eUICC, as shown in Park’s Figure 1 below. Ex-1005, 297; Ex-1002 ¶¶82-83.

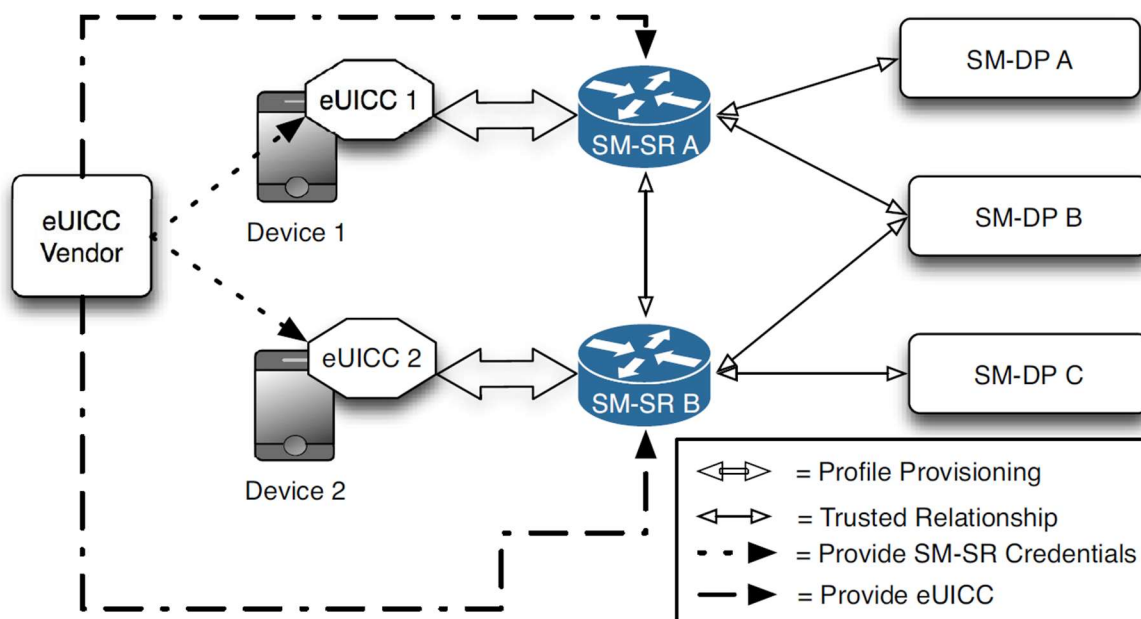


Fig. 1. eUICC Profile Ecosystem

By splitting the SM, Park allows for SM-SR, which securely routes the encrypted profile to the eUICC, to exist in a different trust zone⁶ and potentially be operated

⁶ “Because the profile is security sensitive to the SM-DP itself and the customer, it is obvious that each SM-DP is hardly willing to reveal their generated profile to others even SM-SR.” Ex-1005, 297.

by a different entity than SM-DP and the MNO, thereby allowing SM-DP to negotiate profile encryption keys with the eUICC without exposing them to the SM-SR or other intermediaries. Ex-1005, 299-301. Thus, a POSITA would have understood that Park's SM-DP would maintain control of the encryption of critical data like secret key K, and SM-SR would deliver the encrypted profile to the eUICC but would not have access to the symmetric key, thereby solving the security problem while freeing the MNO from having to directly manage key delivery to each eUICC. Ex-1002 ¶¶82-83, 226.

In this combined system, a POSITA also would have recognized advantages to continuing to encrypt secret key K with a symmetric key that is separate from the key used to encrypt the overall profile. Ex-1002 ¶¶55-56, 227.

First, in embodiments of Nix175's system in which the profile key (the symmetric key used to encrypt the overall profile) is generated dynamically through a key-exchange protocol with the eUICC, the SM would not be able to pre-encrypt the profile with the profile key. At the same time, a POSITA would have understood that sensitive data, such as secret key K, should not be stored in Park's SM-DP in plaintext. Ex-1002 ¶228 (citing Ex-1023, NIST SP 800-57, 73). Thus, encrypting key K under a separate symmetric key would help ensure that Park's SM-DP could store and handle key K securely, before establishing the profile key with the eUICC. Ex-1002 ¶228.

Second, a POSITA would have appreciated that retaining key K in its already-encrypted form, once the profile key has been established, and simply encrypting the entire profile (including already-encrypted key K) with that profile key, would provide additional security benefits, by providing greater protection against compromise during transport. Ex-1002 ¶229 (citing Ex-1024, NIST SP 800-38F, 2). Indeed, Nix175 expressly contemplates this kind of layered encryption. Ex-1016, 33:60-63 (“ciphertext 208b could optionally be included within ciphertext 208a”), 37:57-67 (“a profile 107c can include multiple layers of ciphering”).

This same principle is taught by GlobalPlatform, which teaches first encrypting sensitive data within a message “us[ing] the relevant data encryption session key (DEK)” and then further encrypting the message with the secure channel session key, providing layered protection for these highly sensitive materials. Ex-1006, 264, 258 (explaining that the session key, C-ENC, is used to encrypt “the command data field of the command message being transmitted to the card,” which “includes any data within the data field that has *already been protected* for another purpose *e.g. secret or private keys encrypted with the data encryption session key*”). This configuration is illustrated below in annotated Figure F-11.

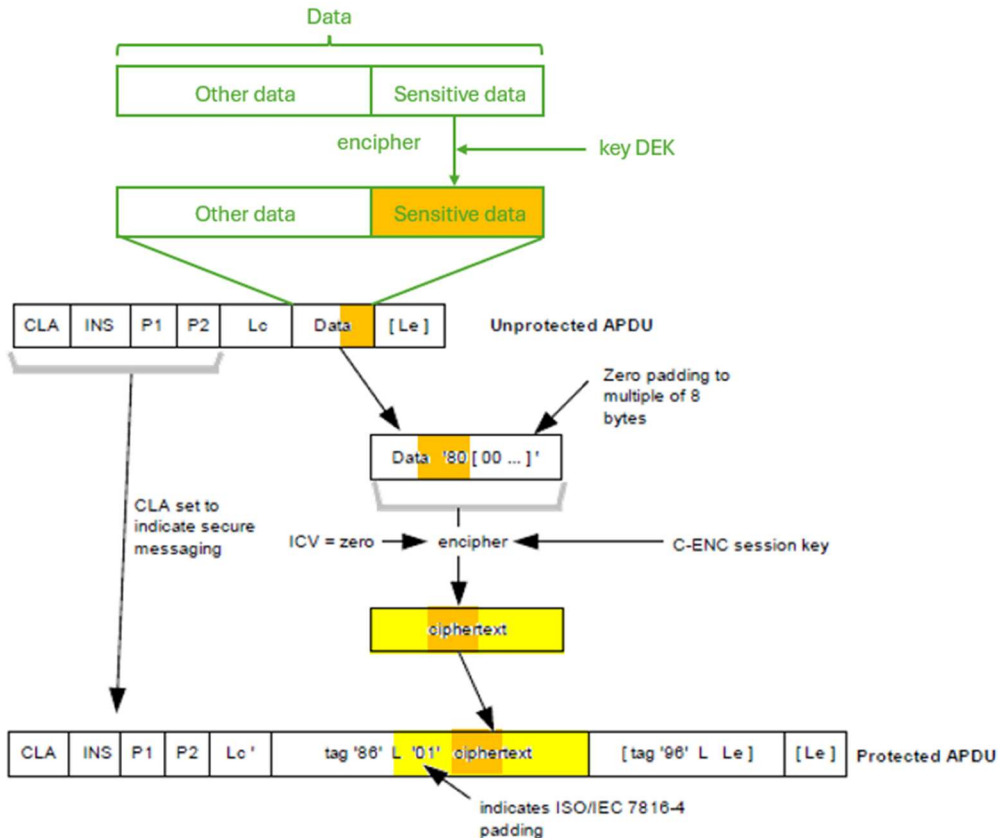


Figure F-11: Secure Messaging: Command message protected for confidentiality

Ex-1002 ¶230.

Thus, in the combined system, profiles sent from the SM-DP to the eUICC would include an inner layer of encryption for sensitive materials like keys (using GlobalPlatform’s DEK), as well as an outer layer of encryption using Nix175’s profile key. Ex-1002 ¶231.

2. Independent Claim 1

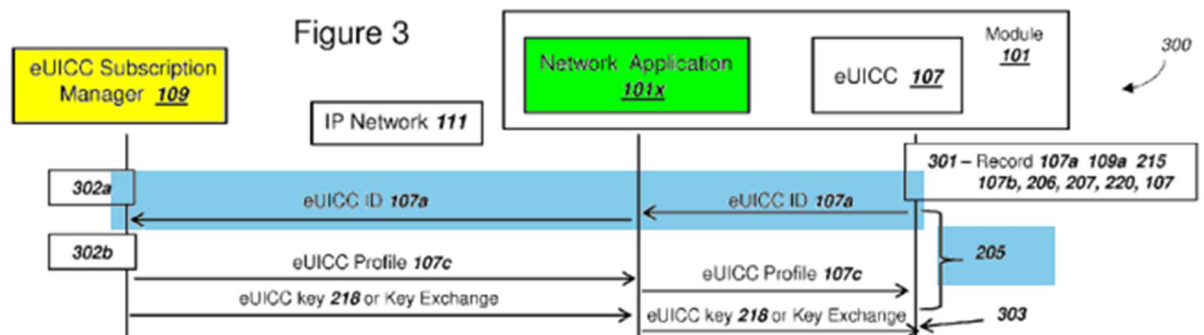
a. Element 1[pre]: A method comprising:

Nix175 teaches “[a] method for authentication, the method performed by a hardware module using an embedded Universal Integrated Circuit Card (eUICC).”

Id. 48:65-49:6. Nix175 explains that “eUICC private key 215 [] can be associated with an eUICC public key 214,” providing the specific example where “eUICC private key 215 and eUICC public key 214 [] comprise a public key infrastructure (PKI) key pair.” *Id.* 41:40-44. Nix175 teaches that the subscription manager stores a copy of the eUICC’s public key (*id.*, 47:52-55), and, in response to receiving this message, uses it to authenticate the eUICC, by verifying the eUICC’s digital signature. Ex-1016, 43:6-19, 49:15-19; Ex-1002 ¶233.

c. Element 1[b]: (b) sending the first message via a network application to a subscription manager system;

As shown in Figure 3, Nix175 shows that the message in step 205 is sent via a network application 101x (green) to a subscription manager 109 (yellow):



Ex-1016, Fig. 3; Ex-1002 ¶234.

- d. **Element 1[c]: (c) deriving a second eUICC private key and a corresponding second eUICC public key using a first random number generator and a first set of cryptographic algorithms;**

Nix175 discloses that the eUICC private/public key pair may be derived using a random number generator and cryptographic algorithms: “Several possibilities exist for the source of an eUICC private key 215 and eUICC public key 214. The *eUICC private key* 215 and *eUICC public key* 214 be generated using standard software tools.” Ex-1016, 42:11-14. “The module 101 could derive the PKI key pair using *a random number generator 128* and a set of *cryptographic algorithms 141*, where the random number generator 128 uses input from a sensor 101f and/or a clock 160 in order to obtain a random number with a high degree of information entropy.” *Id.* 42:25-30; Ex-1002 ¶235.

Nix fails to disclose a second private/public eUICC key pair (different from eUICC private key 215 used for the digital signature in element 1[a]). But a POSITA would have recognized security advantages to deriving a second eUICC private/public key pair for use in the ECDH key-exchange (as described below for Element 1[c][5][i]), separate from the eUICC private/public key pair used for authentication (as described for Element 1[c][1]). Ex-1002 ¶236. In particular, it would have been well known to a POSITA that a key should only be used for a single

purpose, to limit its exposure risk. Ex-1002 ¶236 (citing Ex-1023, NIST SP 800-57, 45 (“In general, a single key **should** be used for only one purpose.”)).

- e. **Element 1[d]: (d) storing a subscription manager public key which corresponds to a subscription manager private key;**

Nix175 teaches “As illustrated in FIG. 2d, the eUICC subscription manager 109 may also be associated with an eUICC *subscription manager public key 220* and an *eUICC Subscription manager private key 222*, and the two keys can comprise a *PKI key pair* for the eUICC subscription manager 109.” Ex-1016, 43:20-24. Nix175 further teaches “module 101 can record an eUICC identity 107a, an address of an eUICC subscription manager 109, and an eUICC private key 215.” *Id.* 47:15-17. And “[o]ther data could be included in a step 301 as well, such as, but not limited to, ... an eUICC public key 214, [and] an *eUICC subscription manager public key 220*.” *Id.* 47:22-25. Ex-1002 ¶237. As explained for Element 1[c][5][i], below, the subscription manager public key is used in a key exchange, to derive the profile key. In the combined system, in which Park’s SM-DP is trusted with encrypting the profile, Nix175’s eUICC subscription manager public key would correspond to the SM-DP public key. Ex-1002 ¶237.

- f. Element 1[e][1]: (e) deriving a profile key using a key exchange algorithm based on at least:**
- (i) the second eUICC private key, and**
 - (ii) the stored subscription manager public key,**

Nix175 teaches “An eUICC 107 within a module 101 could use a ECDH key exchange in a step 303 when ECC algorithms are utilized for eUICC public key 214, eUICC private key 215, and eUICC Subscription manager private key 222 and eUICC Subscription manager public key 220.” Ex-1016, 51:65-52:2. Nix175 refers to a 2013 Wikipedia article for a description of the ECDH algorithm (*id.* 52:4-5), and it was well known that an ECDH algorithm derives a shared key (“profile key”) based on A’s private key and B’s public key⁷. For example:

Alice selects a random number n_a and computes $Y_a = n_a B$, she sends Y_a to Bob. Similarly Bob computes $Y_b = n_b B$ and sends Y_b to Alice. Alice and Bob generate the same key $K = n_a Y_b = n_b Y_a = n_a n_b B$.

Ex-1007, 3. The derived key K in this example from Abi-Char corresponds to the claimed “profile key” calculated from A’s (eUICC’s) private key n_a (eUICC private key) and B’s (subscription manager’s) public key Y_b (subscription manager public key). Ex-1002 ¶238.

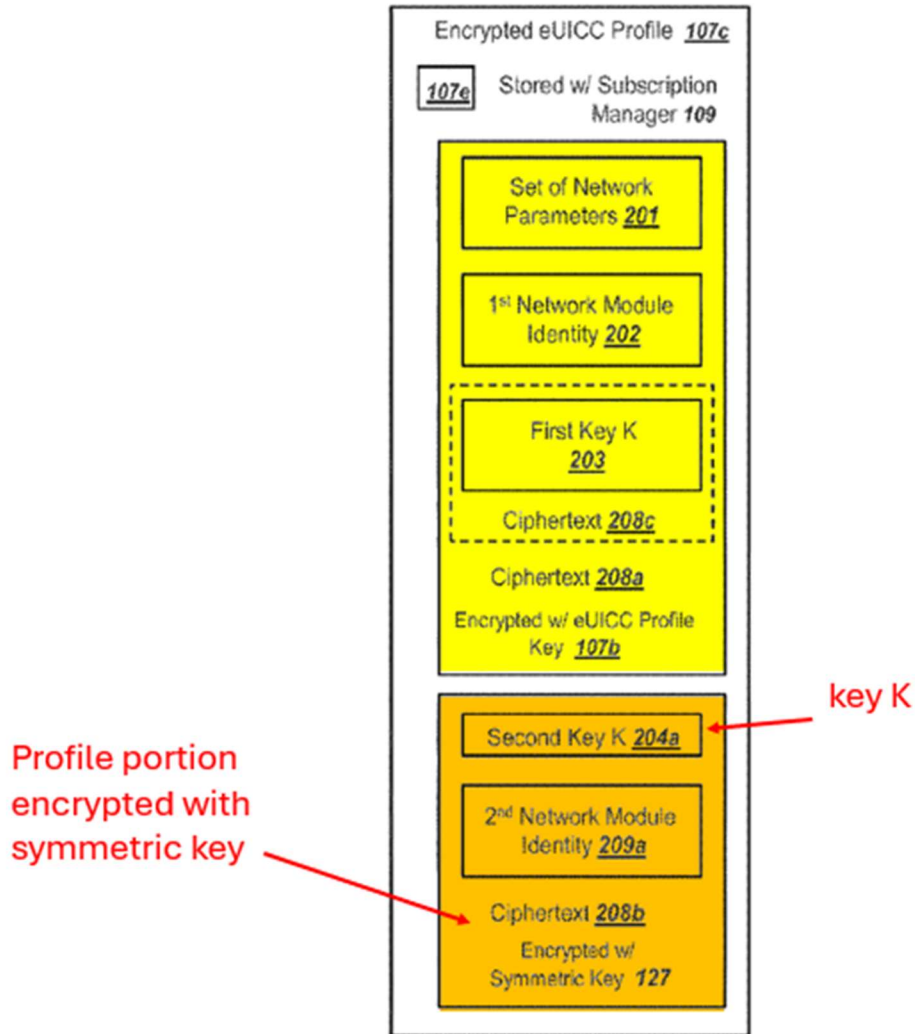
⁷ For example, Nix175 cites X-9.63 as describing an exemplary key exchange. Ex-1016, 52:22-23; *see* Ex-1008, 12 (describing DH key exchange).

- g. Element 1[e][2]: wherein the profile key can also be derived at the subscription manager system based at least on: the second eUICC public key, and the subscription manager private key;**

As discussed above, the same profile key can also be calculated by B (subscription manager) using B's private key n_b (subscription manager private key) and A's public key Y_a (eUICC public key), such that $K = n_a Y_b = n_b Y_a = n_a n_b B$. Ex-1007, 3. Thus, the same profile key can also be derived by the subscription manager using its private key and the eUICC public key. Ex-1002 ¶239.

- h. Element 1[f]: (f) receiving from the subscription manager system, an encrypted profile comprising a ciphertext including a key K encrypted with a symmetric key;**

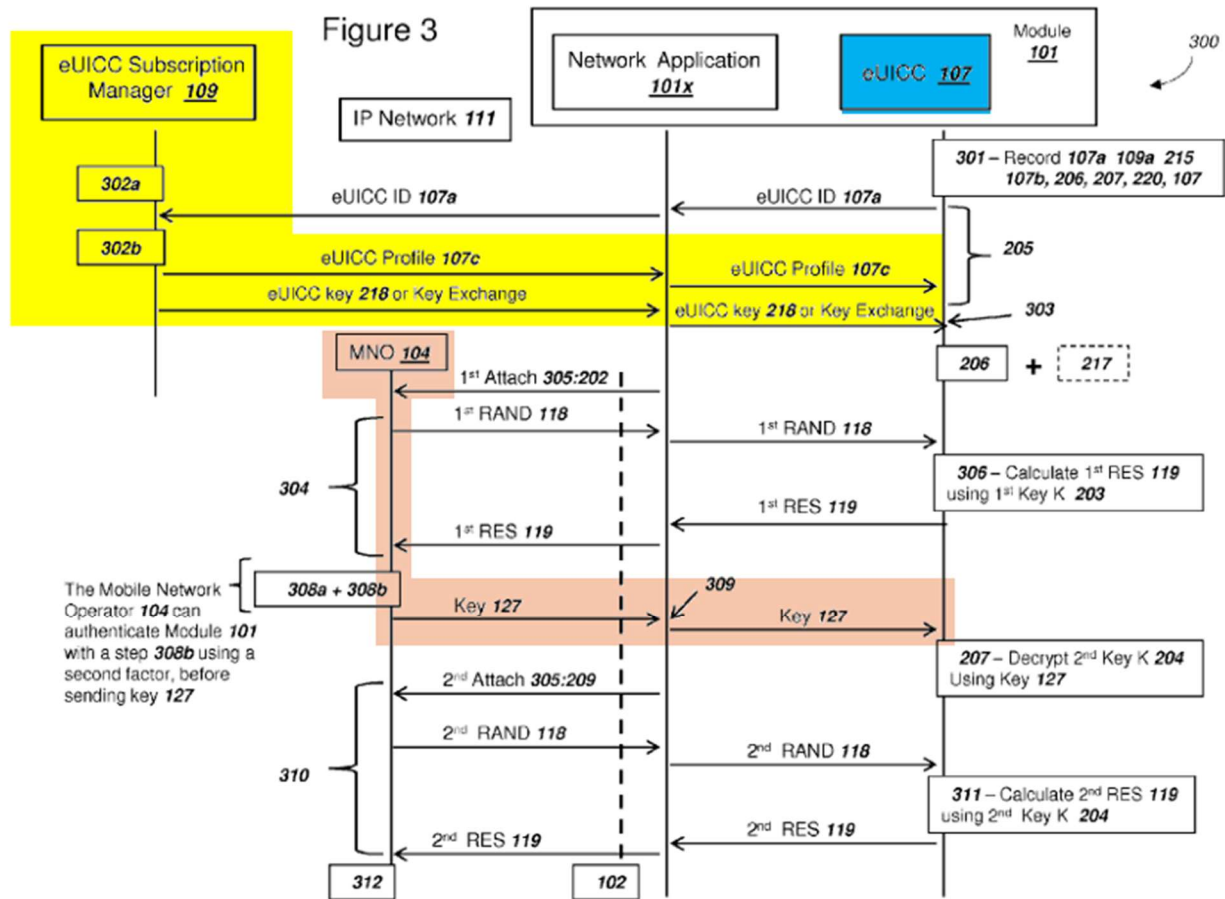
Nix175 teaches “receiving, by the hardware module, a profile from the eUICC subscription manager ... wherein the hardware module uses the symmetric key to decrypt the second portion of the profile, and wherein the second portion includes a second key K2.” Ex-1016, Cl. 1. The portion of the profile encrypted with the symmetric key and comprising ciphertext 208b including “second key K 204a” (the claimed “key K”), is shown in orange in the annotated excerpt of Figure 2a, below.



Ex-1016, Fig. 2a, 40:9-13. Ex-1002 ¶240.

i. Element 1[g]: (g) receiving the symmetric key;

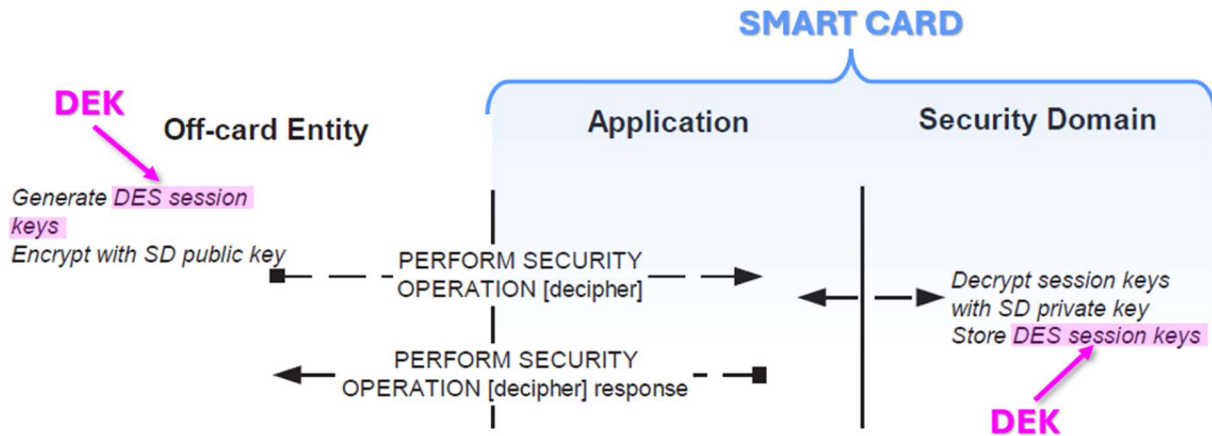
As discussed above, Nix175 teaches that the *MNO* sends the symmetric key 127 used to encrypt the key K in the profile, as shown below in red.



But it would have been obvious, in light of Park-GlobalPlatform, that the subscription manager could send the symmetric key 127 to the eUICC, provided that Park’s split architecture of an SM-SR and SM-DP were used to address the security issue discussed above in Section XII.D.1; Ex-1002 ¶¶241-42.

Specifically Park-GlobalPlatform teaches a “key transport option” that may be used by an off-card entity to send DEK (equivalent to Nix175’s “symmetric key”) to a smart card. Ex-1006, 251-52. The DEK is sent to the card using the “PERFORM SECURITY OPERATION [decipher] command.” Ex-1006, 255-56 (explaining that

the PERFORM SECURITY OPERATION [decipher] command includes “control reference templates” that “contain the actual key values,” with “control reference template ‘B8’” used to transmit the DEK). Ex-1006, 247, 275-76. This is illustrated in Figure F-9, excerpted below:



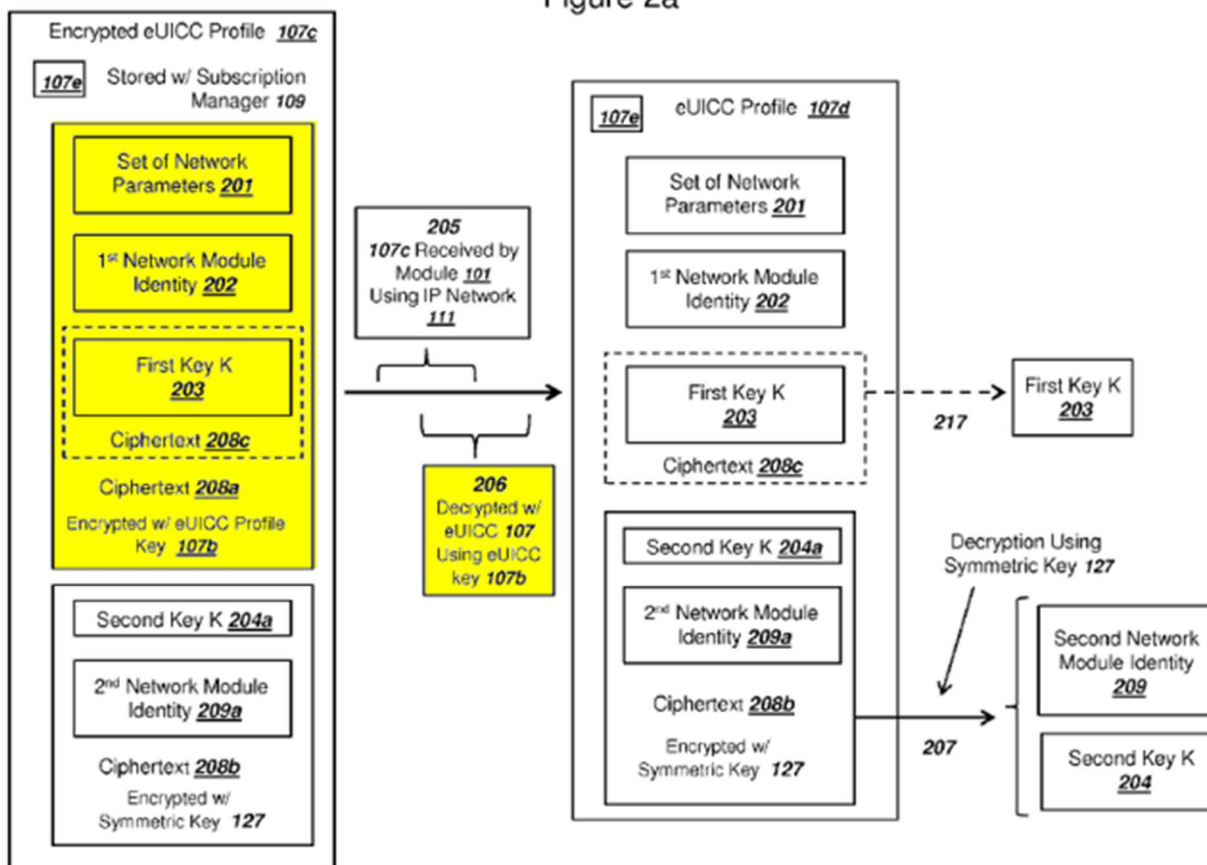
Ex-1002 ¶243.

A POSITA would have recognized that in the combined Nix175-Park-GlobalPlatform system, Park’s SM-DP would send the DEK (claimed “symmetric key”) to the eUICC. Ex-1002 ¶¶40-41, 244. Further, the SM-DP is part of the network trust zone separate from the SM-SR. Ex-1002 ¶¶82-83. Thus, a POSITA would have recognized that in the Nix175-Park-GlobalPlatform system, the subscription manager (component SM-DP) would send the symmetric key to the eUICC without rendering void its security enhancements (as would be the case if the subscription manager of the unmodified Nix175 were to send it). Ex-1002 ¶244.

- j. **Element 1[h]: (h) decrypting, by the embedded universal integrated circuit card, at least a portion of the encrypted profile using the profile key;**

Nix175 teaches “receiving, by the hardware module, a profile from the eUICC subscription manager, wherein the hardware module uses the *profile key to decrypt a first portion of the profile.*” Ex-1016, Cl. 1. This is illustrated in the annotated Figure 2a, below, showing the portion of the profile encrypted by the profile key (“eUICC profile key 107b”) in yellow and being decrypted with that key in step 206. Ex-1016, Fig. 2a; Ex-1002 ¶245.

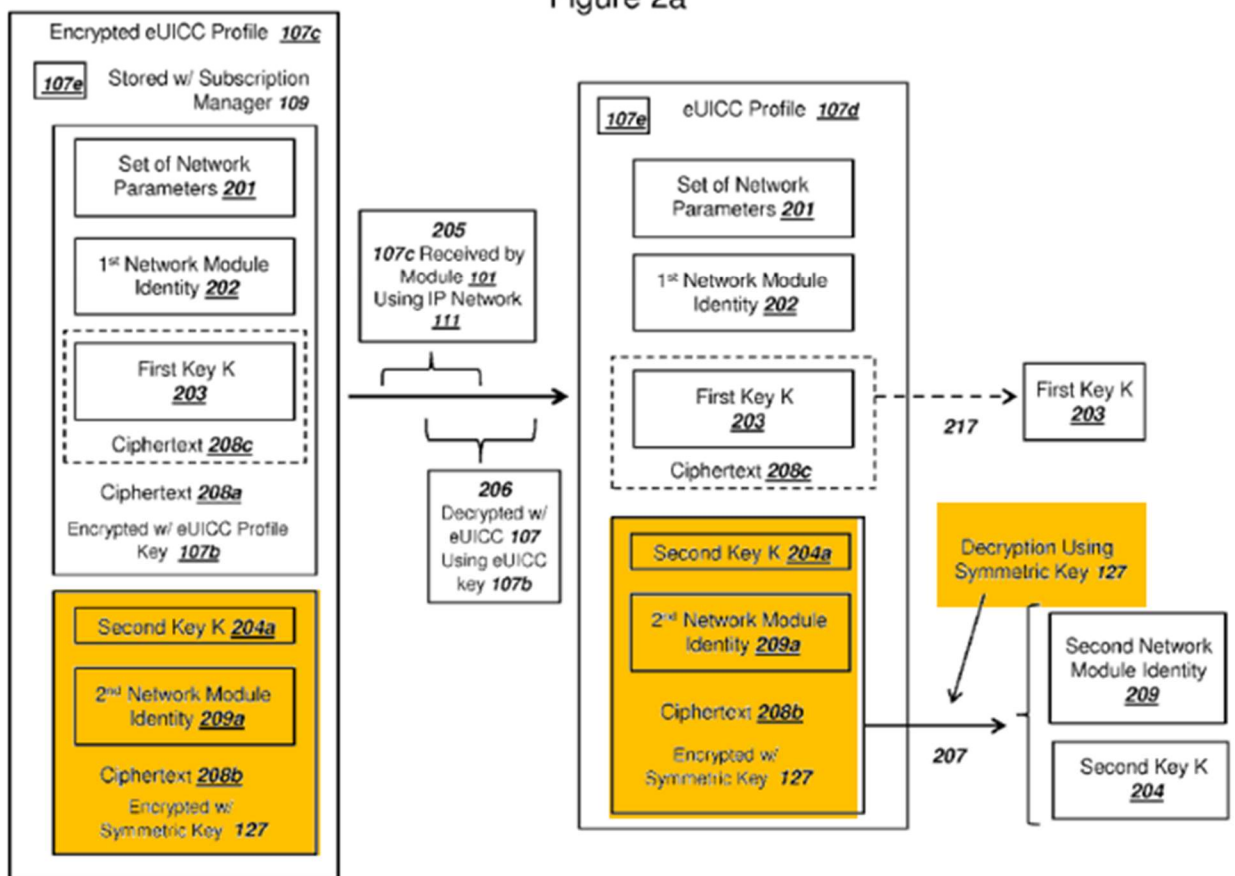
Figure 2a



- k. Element 1[i]: (i) decrypting, by the embedded universal integrated circuit card, at least a portion of the ciphertext using the symmetric key; and

Nix175 teaches “wherein the hardware module *uses the symmetric key* to *decrypt the second portion of the profile*, and wherein the second portion includes a second key K2.” Ex-1016, Cl. 1. This is illustrated in Figure 2a below, where orange shading shows the second portion including key 204a encrypted by symmetric key 127 and then being decrypted by symmetric key 127 at step 207. Ex-1016, Fig. 2a; Ex-1002 ¶246.

Figure 2a



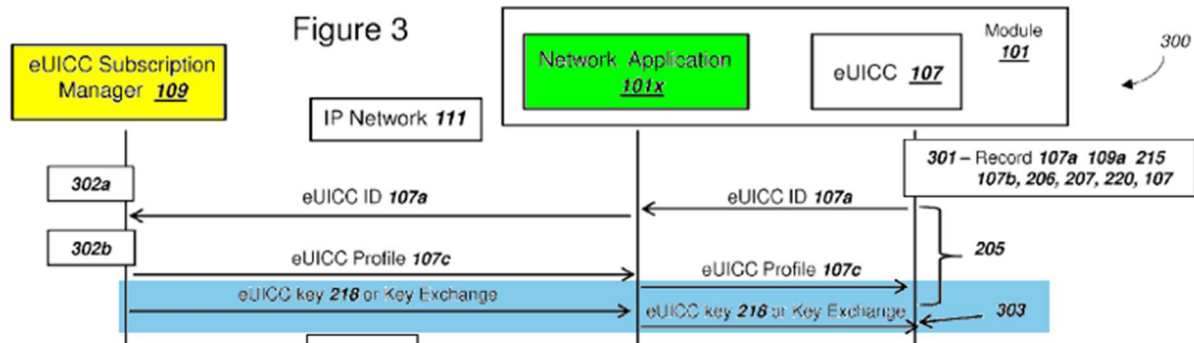
l. Element 1[j]: (j) storing at least the key K in the embedded universal integrated circuit card for use in future communications.

Nix175 teaches that the “second key K 204 can also be used with standards-based authentication with a wireless network 102, where the second key K 204 in FIG. 2a is also depicted and described as operating as a standards-based shared secret key K in FIG. 3 below at a step 311.” Ex-1016, 32:31-35. Further, Nix175 teaches “wherein the UICC records the second key K2 in a protected nonvolatile memory after the eUICC decrypts the second key K2.” *Id.* Cl. 8. Thus, a POSITA would have understood the claimed “key K” would have been stored in the eUICC for use in future communications, according to the wireless communication standards. Ex-1002 ¶247.

3. Dependent Claims 2-22

a. Claim 2: The method of claim 1, wherein the subscription manager public key is downloaded via the network application.

Nix175 teaches that the “ECDH key exchange in a step 303,” illustrated below, includes “eUICC subscription manager public key 220.” Ex-1016, 51:65-52:2.



Nix175 further teaches that this subscription manager public key is “recorded with an eUICC 107 in a step 301” (*id.*, 49:49-55), which may occur at the time of manufacture, or, alternatively, “the data could be downloaded or accessed separately via an IP network 111.” *Id.*, 47:18-22. A POSITA would have understood that this download would occur from the subscription manager (yellow) via network application 101x (green). Ex-1016, Fig. 3, 46:7-29. Ex-1002 ¶¶248-49.

- a. **Claim 3: The method of claim 1, wherein the step of storing the subscription manager public key occurs at the time of manufacture of the embedded universal integrated circuit card.**

Nix175-Park-GlobalPlatform teaches Claim 3. Ex-1002 ¶250. As explained for Claim 2, above, Nix175 teaches that the subscription manager public key (corresponding to Park’s SM-DP public key in the combined system) is “recorded with an eUICC 107 in a step 301” (*id.*, 49:49-55), which may be performed by a “manufacturer” at the time of manufacture. *Id.*, 47:18-22. Similarly, Park teaches that the “eUICC Vendor develops eUICCs. During this, the SM-SR credentials (key_{SM-SR}) and the capability information ($CI_{installed}$) of each eUICC are installed into

each eUICC.” Ex-1005, 300. A POSITA would have understood that, in the combined system, Park’s eUICC Vendor would store both SM-SR credentials and the SM-SD credentials (in the form of an SM-SD public key). Ex-1002 ¶250.

b. Claim 4: The method of claim 1, wherein the key exchange algorithm is a Diffie-Hellman key exchange.

Claim 5: The method of claim 1, wherein the key exchange algorithm is an Elliptic Curve Diffie-Hellman key exchange.

Nix175 teaches that “An eUICC 107 within a module 101 could use a ECDH [Elliptic Curve Diffie Hellman] key exchange in a step 303 where ECC algorithms are utilized for eUICC public key 214, eUICC private key 215, and eUICC subscription manager private key 222 and eUICC subscription manager public key 220.” Ex-1016, 51:65-52:2; Fig. 3. Thus Nix175 teaches the key exchange algorithm is a Diffie-Hellman exchange and specifically, an elliptic curve Diffie-Hellman exchange. Ex-1002 ¶¶82-83, 251.

c. Claim 6: The method of claim 1, wherein the encrypted profile is encrypted in a manner so that intermediate nodes on an IP network would not be able to read data within the profile in an unencrypted form.

Nix175 teaches “eUICC subscription manager 109 sends the profile 107c, where the network access credentials (including the first key K 203) are encrypted with an eUICC profile key 107b, and in this manner intermediate nodes on the IP

network 111 would not feasibly be able to read the data within the profile 107c.” Ex-1016, 51:24-28. Ex-1002 ¶252.

d. Claim 7: The method of claim 1, wherein subscription manager system stores the ciphertext and the symmetric key before step (a).

Nix175-Park-GlobalPlatform teaches Claim 7. Ex-1002 ¶¶253-54.

Park’s SM-DP prepares the profiles to be sent to the eUICC, and as explained in Section **Error! Reference source not found.** of Ground 1, GlobalPlatform taught to separately encrypt sensitive key data with DEK. Because Park’s SM-DP can encrypt a portion of the profile with DEK (the claimed “symmetric key”) without needing to know the eUICC destination, it can do so, and store DEK, before step (a), when the eUICC sends its eUICC ID. Ex-1002 ¶254. It would have been a simple matter of design choice of whether to pre-encrypt profiles with the symmetric key to speed the process of provisioning at the cost of increased storage requirements in the SM-DP. Ex-1002 ¶254. Moreover, Nix175 teaches “eUICC subscription manager 109 could assemble and cipher the profile 107d into a profile 107c at an earlier time, such as before receiving the eUICC identity 107a.” Ex-1016, 50:57-59; Ex-1002 ¶254.

- e. **Claim 8: The method of claim 1, wherein after the step (b), the embedded universal integrated circuit card is authenticated by the subscription manager system using at least the identity of the embedded universal integrated circuit card included in the first message.**

Claim 9: The method of claim 1, wherein after the step (b), the embedded universal integrated circuit card is authenticated by the subscription manager system using at least the digital signature included in the first message.

Nix175 teaches “An eUICC subscription manager 109 can take several actions in a step 302b after receiving an identity for the eUICC 107 or module 101. A first action in a step 302b could comprise *authenticating an eUICC 107* or a module 101 *based on the eUICC identity 107a received in the message*. ... In addition, the eUICC Subscription manager could *authenticate using a digital signature algorithm 221*, where the message with the eUICC identity 107a could *include a digital signature*, as described in the paragraph above.” Ex-1016, 49:7-19. Ex-1002 ¶255.

- f. **Claim 10: The method of claim 1, wherein the decrypted ciphertext is stored in nonvolatile memory of the embedded universal integrated circuit card.**

Nix175 teaches that the ciphertext decrypted to obtain the “second key K 204” is stored in the eUICC’s nonvolatile memory: “The module 101 or eUICC 107 can record the plaintext second key K 204 in a nonvolatile memory such as, but not limited to, flash memory 101 w such that the plaintext second key K 204 remains

available to module 101 after power or a battery 101 k is removed from module 101.” Ex-1016, 59:1-5; Ex-1002 ¶256.

- g. Claim 11: The method of claim 1, wherein the embedded universal integrated circuit card is operably connected to a module and wherein the module is a machine-to-machine module.**

Nix175 teaches “In a first exemplary embodiment for a step 308b in FIG. 3 where module 101 supports M2M applications with a M2M service provider 115, MNO 104 could securely send to M2M service provider 115 an identity of module 101, such as, but not limited to module identity 110, eUICC identity 107a, profile identity 107e, and/or the first network module identity 202 through an IP network 111.” Ex-1016, 57:14-21. Ex-1002 ¶257.

- h. Claim 12: The method of claim 11, wherein the module comprises a module of a type selected from the group consisting of: vending machine, automobile, alarm system, tracking device, monitoring device, security system and remote control device.**

Nix175 teaches “Among many potential benefits, M2M technologies allow the remote monitoring and/or control of people, assets, or a location where manual monitoring is not economic, or costs can be significantly reduced by using automated monitoring as opposed to manual techniques. Prominent examples today include vending machines, automobiles, alarm systems, and remote sensors.” Ex-1016, 1:27-33. Ex-1002 ¶258.

- i. **Claim 13: The method of claim 1, wherein the embedded universal integrated circuit card is operably connected to a module and wherein the module is a wireless module.**

Claim 14: The method of claim 13, wherein the module is configured to communicate using one or more wireless technologies of types selected from the group consisting of: WiFi, WiMax, General Packet Radio Services, Enhanced Data rates for GSM Evolution and 3rd Generation Partnership Project technology.

Nix175 teaches “[a]n eUICC 107 within module 101” (Ex-1016, 11:36), where module 101 may operate “as a wireless module” that can communicate with a wireless network using a variety of wireless technologies, include “WiFi, WiMax, a 2nd generation wireless wide area network (WAN) technology such as, but not limited to, General Packet Radio Services (GPRS) or Enhanced Data rates for GSM Evolution (EDGE), [or] 3rd Generation Partnership Project (3GPP) technology such as, but not limited to, 3G, 4G LTE, or 4G LTE Advanced.” Ex-1016, 9:45-53; Ex-1002 ¶259.

- j. **Claim 15: The method of claim 13, wherein the module comprises a module of a type selected from the group consisting of the following: wireless handset, cellular phone, smartphone, laptop computer, tracking device, computer with a radio, circuit board with a radio, hand-held device, netbook, portable computer, multiprocessor system, microprocessor based consumer electronic, programmable consumer electronic, network personal computer, minicomputer, mainframe computer and smartcard.**

Nix175 teaches “module 101 can be a wireless handset, a cellular phone, a smartphone, a tablet computer, a laptop, a computer with a radio, a tracking device, or a circuit board with a radio that accesses wireless network 102.” Ex-1016, 10:25-29. Ex-1002 ¶260.

- k. **Claim 16: The method of claim 1, wherein the embedded universal integrated circuit card is operably connected to a module, wherein the module is a wired module.**

Claim 17: The method of claim 16, wherein the module is configured to communicate using one or more wired connections of types selected from the group consisting of: Ethernet, fiber optic connection and Universal Serial Bus connection.

Nix175 teaches “A wireless or a wired configuration for module 101 can be utilized in the present invention.” Ex-1016, 9:42-44. When module 101 is a wired module, Nix175 teaches that it can connect to the IP Network “via a wired

connection such as, but not limited to, an Ethernet, a fiber optic, or a Universal Serial Bus (USB) connection.” Ex-1016, 9:55-58. Ex-1002 ¶261.

- l. Claim 18: The method of claim 16, wherein the module comprises a module of a type selected from the group consisting of: computer, security camera, security monitoring device and networked controller.**

Nix175 teaches “In a wired configuration (not shown), module 101 can be a computer, security camera, security monitoring device, networked controller, etc.” Ex-1016, 9:42-44. Ex-1002 ¶262.

- m. Claim 19: The method of claim 1, wherein the deriving step (c) is performed prior to the generating step (a).**

Claim 20: The method of claim 1, wherein the generating step (a) is performed before the deriving step (c).

Nix175 teaches for step (c) that “eUICC private key 215 can be processed or derived using a random number.” Ex-1016, 41:63-64. Because the message generated in step (a) is independent of the eUICC private key generation process, it would have been a simple matter of design choice to perform step (c) before step (a) or vice versa. Ex-1002 ¶263.

- n. **Claim 21: The method of claim 1, wherein the embedded integrated circuit card is operably connected to a module and a user accesses a user interface of the module to enter authentication information before the module receives the symmetric key.**

Nix175 teaches “the module 101 receives the symmetric key 127 *after* the MNO 104 authenticates or verifies that a user 113 or M2N service provider 115 is associated with module 101, and until the receipt of the symmetric key 127 the cipher text 208b cannot be feasibly decrypted.” Ex-1016, 58:45-49. Further “wherein a user accesses *a user interface* of the hardware module to authenticate with the mobile network operator *before the hardware module receives the symmetric key.*” *Id.* Cl. 2. Ex-1002 ¶264.

In the combined system, in which the subscription manager is split into Park’s SM-SR (not necessarily trusted by the MNO) and SM-DP (trusted), the SM-DP would be tasked with authenticating the user, before sending either the symmetric key or the encrypted profile to the user. Ex-1002 ¶265.

- o. **Claim 22: The method of claim 1, wherein the subscription manager receives the ciphertext from a mobile network operator before the network application sends the first message.**

As discussed above for claim 7, Nix175 teaches “eUICC subscription manager 109 could assemble and cipher the profile 107d into a profile 107c at an

earlier time, such as before receiving the eUICC identity 107a.” Ex-1016, 50:57-59;
Ex-1002 ¶266.

XIII. CONCLUSION

Petitioner requests institution of IPR for Claims 1-22 of the '094 Patent based on each of the grounds in this Petition.

Respectfully Submitted,

/s/ William M. Fink

William M. Fink (Reg. No. 72,332)

CERTIFICATE OF WORD COUNT

Pursuant to 37 C.F.R. §42.24(d), Petitioner certifies that this petition includes 13,865 words, as measured by Microsoft Word, exclusive of the table of contents, mandatory notices under §42.8, certificates of service, word count, and exhibits.

CERTIFICATE OF SERVICE (37 C.F.R. §42.6(e)(1))

The undersigned hereby certifies that the above document was served on November 26, 2025, by filing this document through the Patent Trial and Appeal Board P-TACTS System, as well as delivering a copy via express mail upon the following attorneys of record for the Patent Owner:

Amster, Rothstein, & Ebenstein LLP
405 Lexington Avenue
Floor 48
New York, NY 10174

A courtesy copy was sent to the below counsel via electronic mail:

Demetrios Anaipakos
Amir H. Alavi
Michael McBride
Steven Jugle
C. Ryan Pinckney
Connie Flores Jones
ALAVI & ANAIPAKOS PLLC
609 Main Street, Suite 3200
Houston, Texas 77002
Telephone: (713) 751-2362
Facsimile: (713) 751-2341
danaipakos@aatriallaw.com
aalavi@aatriallaw.com
mmcbride@aatriallaw.com
sjugle@aatriallaw.com
rpinckney@aatriallaw.com
cfloresjones@aatriallaw.com

Michael F. Heim
Eric J. Enger
R. Allan Bullwinkel
William B. Collier, Jr.
HEIM, PAYNE & CHORUSH, LLP
609 Main St. Suite 3200
Houston, Texas 77002
Telephone: (713) 221-2000
Facsimile: (713) 221-2021
mheim@hpcllp.com
eenger@hpcllp.com
abullwinkel@hpcllp.com
wcollier@hpcllp.com

Andrea L. Fair
Claire A. Henry
MILLER FAIR HENRY PLLC
1507 Bill Owens Parkway
Longview, Texas 75604
(903) 757-6400 (telephone)
andrea@millerfairhenry.com
claire@millerfairhenry.com

Dated: November 26, 2025

Respectfully submitted,

/s/ William M. Fink

William M. Fink (Reg. No. 72,332)

O'Melveny & Myers LLP

1625 Eye Street, NW

Washington, DC 20006

Telephone: (202) 383-5300

Fax: (202) 383-5414

Email: tfink@omm.com

Attorney for Petitioner